1964

A guide to the Stanford Research Institute's inventory management simulator/evaluator with optional probablistic demand patterns.

Gillespie, James A.

Monterey, California : Naval Postgraduate School

http://hdl.handle.net/10945/12273

A GUIDE TO THE STANFORD RESEARCH INSTITUTE'S INVENTORY MANAGEMENT SIMULATOR/EVALUATOR WITH OPTIONAL PROBABILISTIC DEMAND PATTERNS

JAMES A. GILLESPIE

56

[redacted]

A GUIDE TO THE STANFORD RESEARCH INSTITUTE'S INVENTORY MANAGEMENT SIMULATOR/EVALUATOR WITH OPTIONAL PROBABILISTIC DEMAND PATTERNS

* * * * *

James A. Gillespie

A GUIDE TO THE STANFORD RESEARCH INSTITUTE'S INVENTORY MANAGEMENT SIMULATOR/EVALUATOR WITH OPTIONAL PROBABILISTIC DEMAND PATTERNS

by

James A. Gillespie
Lieutenant Commander, Supply Corps, United States Navy

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE
IN
OPERATIONS RESEARCH

United States Naval Postgraduate School
Monterey, California

1 9 6 4

A GUIDE TO THE STANFORD RESEARCH INSTITUTE'S INVENTORY MANAGEMENT SIMULATOR/EVALUATOR WITH OPTIONAL PROBABILISTIC DEMAND PATTERNS

by

James A. Gillespie

This work is accepted as fulfilling

the thesis requirements for the degree of

MASTER OF SCIENCE

IN

OPERATIONS RESEARCH

from the

United States Naval Postgraduate School

ABSTRACT

Simulation of inventory systems is rapidly becoming an accepted management tool for quantifying proposed decision rules. The Stanford Research Institute's Inventory Management Simulator/Evaluator is an excellent example of a multi-branch, multi-item simulator. It represents one of the most sophisticated and comprehensive programs available today. In this Guide, the simulator is completely documented in such a way that anyone with a general knowledge of supply systems can readily follow the rationale of the program. For those who have a knowledge of the FORTRAN language and of programming techniques, a section is devoted to a card-by-card description of the program. Operating instructions and output reports are also discussed in detail. As an added feature, new sub-routines have been designed to produce random demand patterns which approximate the Normal, Exponential, Uniform, and Poisson distributions.

PREFACE

My introduction to the Stanford Research Institute's Inventory Management Simulator/Evaluator occurred during a six week field assignment to that organization in the summer of 1963. At the time I reported, Mr. Peter H. Butterfield, Manager of the Industrial Research Division, suggested that I work with Mr. Michael E. Chambreau on the redesign of their inventory simulator. Being interested in the inventory management field and fresh from a year of Operations Analysis studies I gladly accepted Mr. Butterfield's offer. The experience has proven very valuable and I am sure that it will be even more so in the years ahead. In particular, the work of that summer helped to fill that gap which always exists between the theoretical knowledge of the classroom and the day-to-day application of that knowledge in the "real" world. It certainly gave me a better understanding and appreciation for both the theoretical and practical aspects of the operations researcher's work. Learning the art of applying a science is somewhat akin to putting the flesh on a skeleton. You begin to recognize the value of the theoretical frame-work upon which a problem is solved as well as the value of the solution to that problem. It is with the general thought of helping others to bridge this gap that I have written this paper and highly recommend to succeeding students the study of actual research reports such as these listed in the bibliography.

It is my hope that this volume will serve at least three specific purposes: first, to provide a publication which will give a detailed description of the computer program and necessary operating instructions

such that the simulator may be utilized by the students of the Operations Research, the Management, and the Management Data Processing Curriculums in their studies and thesis work; secondly, as a basic document on which to build a machine simulation course for Supply Corps Officers of the Operations Research curriculum which could be substituted for the present War Gaming course now required for these officers; thirdly, to provide a program which may be utilized as a model and source of basic building blocks with which to design more specialized programs.

I would like to express my thanks to Mr. Butterfield for the courtesies he extended me during my visit at the Institute and for permission to use the simulator here at the Naval Postgraduate School. Also, I am indebted to Mike Chambreau for his patience and instruction in programming techniques and to Professor Rex Shudde and Commander S. W. Blandin for their advice and assistance on the preparation of this thesis.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

CHAPTER I

INTRODUCTION

In documenting the Stanford Research Institute's Inventory Management Simulator I have intentionally directed my efforts towards providing a working reference for students of the Naval Postgraduate School. An attempt has been made to explicitly point out the assumptions and ground rules of the program in order to define the system being simulated. It is felt that by clearly defining the system, the simulator may be used by students who have no training in computer language as well as by those who have. I have also attempted to minimize the time required for a student to become thoroughly familiar with the program by presenting as many details as possible without overly boring the reader. The mathematical derivations of the optimal requisition and reorder level equations have not been reproduced in this paper. Students who are interested in reviewing these are referred to reports 1 and 2 of the Bibliography. I highly recommend to the Supply Corps Officers in the Operations Research curriculum the study of SRI's Technical Memorandum No. 3. In addition to providing the basis of the simulator program, it is also an excellent example of Operations Research at work. Chapter III which is a card-by-card description of the FORTRAN Language coding and program logic may be omitted by the student who is not familiar with the FORTRAN.

CHAPTER II

THE SUPPLY SYSTEM TO BE SIMULATED

1. General description

The Supply System for which the simulator is designed, consists of a manufacturing facility, a central or "wholesale" warehouse and a number of branch or "retail" warehouses. The system is intended to be a "push" system with a control office located at the central warehouse which periodically reviews at fixed intervals the system inventory positions and initiates action to resupply the system by placing production orders or resupplies a particular location by making interim shipments from the central warehouse stocks. There is no provision made to relocate excess stock at one branch to another which has a deficiency.

The physical distribution of stock to the branches is made from the central warehouse which is located at the manufacturing center. Production orders are coordinated with the expected needs of the branches and allocations are made to the branches at the time of completion of the manufacturing process. The production quantity is physically delivered to the central warehouse which immediately computes the need of each branch based on its current inventory position and makes direct allocations as required. Shipments are made to each branch on a fixed schedule depending upon the particular requirements of a branch. There are no provisions made for emergency type deliveries. The central warehouse acts as a branch or "retail" outlet within its region as well as being a stock point for the entire system.

The simulator was originally designed in connection with studies of products which have a common production operation up to a certain point and then are either modified slightly or packaged in different sizes which

then become distinct items of the product in the inventory. These items retain a certain degree of interchangeability with respect to management decisions. There is, however, no provision in the simulator to substitute one item of stock for another to fill customer demands. However in reviewing the system as a whole it is desirable to consider not only the items individually but also to consider the inventory status of the product they comprise. Examples of this type of product are lubricants, paints, glass, plastics, lumber, clothing, gasses, canned foods, dairy products, etc. Even though one particular item (or size) may not be readily substituted for another, there is still a management need to know the total quantity and/or value of the basic product for determining production and inventory policies as well as production quantity. For example, if a clothing manufacturer-distributor wanted to produce polka-dot undershorts in various sizes he would need to know the total quantity of cotton broadcloth to produce first. (In order to accumulate product data the units of an item are converted to the product-unit by a size factor and then sum over all items related to the product.)

The simulator can also be used for systems where this feature does not occur simply by considering each item as a product and setting the size factor to unity.

The SRI Inventory Management Simulator/Evaluator is time oriented and performs the usual functions of forecasting, ordering, distributing, issuing, and record keeping within each time frame of the program. In the present form, the simulator has fifty-two time cycles representing the fifty-two weeks of a year. Within each time frame all the normal inventory functions are carried out in a set sequence which was arbitrarily chosen. This arbitrary sequence of functional operations does not bias

the system since over a long run the system performance will be independent of the actual functional sequence used within a time frame.

The data for one product and all of its related items are processed and massaged completely before beginning simulation of the next product. Hence, there is also a cycling on the various products involved in the simulation to produce the overall system data which is desired in the summary reports.

2. Optimal values for system control rules

The following variables are designed as control variables and form a set of parameters which will provide definite policy rules for the system:

RIJK(J,K) The requisition objective level for the jth item of a product at the kth location. The maximum level in item units to which the inventory position is raised at production time.

XIJK(J,K) The reorder level for the jth item of a product at the kth location. The level in item units below which an interim branch order is placed.

XIJ(J) The system-wide reorder level for the jth item of a product. The level in item units below which a new production order is placed.

XI The system wide reorder level for a product. The level in product units below which a new production order is placed.

XIJK(J,1) The central warehouse reorder level for the jth item of a product. The level in item units below which a new

production order is placed.

QI The minimum economic production order quantity (EOQ).

Since the overall objective of the supply system is to minimize its total cost of operation, optimal settings of the control variables could be obtained by determining each of the cost components (i.e., holding cost, production ordering cost, and shortage cost) as a function of the control variables and then minimizing with respect to all of the variables. However, the dual functioning of the warehouse as a system stock point for interim orders and as a branch outlet unduly complicates the mathematics. To avoid this, the warehouse is considered to be an independent branch with demand rate equal to the sum of the direct demand rate plus the demand rate due to interim branch orders. This reduces our two level system to a one level system with independent branches. The optimal values for the control variables are then computed according to formulas developed in [2]. This approximation seems reasonable since the interim branch orders are not expected to be too great.

Economic Production Order Quantity

The optimal value for the economic order quantity is given in [2],

as:

$$QI = \begin{cases} Qw + f & \text{if } \left[Qw/f\right]^2 \geq 2.25 \\ 1.85\ Qw^{2/3}\ f^{1/3} & \text{otherwise} \end{cases}$$

The "Qw" is the classical or "Wilson"[1] economic order quantity which balances the inventory holding costs against the production ordering cost. The "f" is a correction factor to account for the demands during

1 Wilson, R. H. A New Method of Stock Control. Harvard Business Review, Vol. 5, 1926-1927 pp.197-205.

leadtime and the variance of the demand and leadtime distributions. If the value of "Q" so determine approaches the shelf life limitations then a maximum limit is taken as the demands during a shelf-life period less one and a half times the demands during a production leadtime. This is a rather arbitrary setting for the maximum value. No minimum value is set.

Product Reorder Level

The optimal reorder level for the product is taken as the difference between the optimal requisition level for the product and the economic order quantity. The optimal requisition level for the product is the sum of the optimal levels for the branches. Since reviews of the inventory positions are made at discrete intervals, the time at which the reorder levels are reached will fall randomly within the review interval. Then the expected demands which occur after the reorder level is actually reached and before this fact is recognized will be the demand rate times one-half the review period. Likewise, since the average number of units demanded per requisition is not necessarily unity the inventory position when reviewed will be below the reorder level by one-half of the average number of units per requisition. Hence, the reorder level is to be raised by one-half of the demands during a review period plus one-half of the expected number of units demanded per requisition to compensate for the discrete nature of the review time and demand quantity.

Branch Requisition Objectives (Other Than Central Warehouse)

The optimal requisition objective for an item-location is derived as a function of the demand rate, the total product demand rate, the variances of the demand and leadtime distributions, the production

leadtime, the holding cost and the cost of shortages. The mathematical details are given in [2.]. A maximum and minimum level are set to guard against shelf life limitations and unusual values of the input parameters. The maximum requisition objective level is set at the expected demands during a shelf life period and the minimum is set at the reorder level plus the demands during a production leadtime and a production cycle.

The formula for the optimal requisition objective level is:

$$R_{ijk} = f_{ijk} \ln \left\{ K_{jk}(b^{1}_{ijk}{}^{Q^{1}} - 1)\ (1 + C^{1}_{sjk}/(k_{jk}C_{hjk}L_{jk}))/t \right\}$$

where f_{ijk} is demands during leadtime corrected for the variances of the demand and leadtime distributions

K_{jk} is a variance factor

$(b^{1}_{ijk}{}^{Q^{1}} - 1)$ is a probability factor

C^{1}_{sjk} is a cost of shortages

k_{jk} is another variance factor

C_{hjk} is the holding cost

L_{jk} is the production leadtime

t is the production order cycle

Central Warehouse Requisition Objective

Like the requisition objective levels for the branches, the optimal requisition objective for the central warehouse is determined as a function of the demand rate, the variances of the demand and leadtimes

distributions, the branch reorder leadtime, the holding cost, and the cost of shortages. However, in the case of the central warehouse the situation is complicated by the fact that the total demand is the sum of the direct demands and the branch interim orders. Since the later is a random quantity, it must be computed as an expected value. It is assumed that the "tail" of the distribution of the demands at any branch can be approximated by a negative exponential whose mean is taken to be that of the true distribution. Then the probability of an order occurring times the expected size of an interim shipment given that the need occurs is the unconditional expected demands on the central warehouse from a branch. The details of the computations of the probability of an order occurring and the conditional expected size of an interim order are given in [1.].

The cost of shortages must also be computed as an expected value since the cost of shortages pertaining to non-availability of stock to meet direct demands is not the same as that for demands arising from branch interim orders. For direct demands the cost of shortages is the profit lost. The shortage cost applied to the central warehouse for not being able to fill branch interim orders is computed as a function of the rate of change of the expected shortage quantity in a branch reorder leadtime with respect to the reorder leadtime. As the reorder leadtime increases the probability of a branch shortage increases and hence, the expected shortage at the branch and resultant costs of shortage will increase. The actual cost to be lodged against any warehouse shortage is then the expected value of the direct and indirect demand shortage costs average with respect to the frequency of occurrence or probability distribution of the direct and indirect demands.

As with the requisition objective levels for the branches, a maximum and minimum levels are established for the warehouse to guard against shelf life limitations and unusual settings of the input parameters. The maximum is taken as the total warehouse demands during a shelf life and the minimum is the sum of the reorder level and the total demands during a production order cycle and production order leadtime for the warehouse.

Branch Reorder Level

Although an optimal reorder level formula is derived in [2.] which is analogous to the optimal requisition objective formula, it is not utilized in the simulator. The branch reorder level is taken to be the demands during a branch interim reorder leadtime. The simulator does provide a means of increasing or decreasing this level to study the sensitivity of the overall system to changes in the reorder point.

Central Warehouse and Item Production Reorder Levels

The central warehouse production reorder level is taken to be the total demands on the warehouse during its production leadtime.

The system-wide item production reorder level is the demands during a branch production leadtime summed over all branches including the central warehouse.

Provisions are made in the simulator to adjust the reorder level settings in order to study the effects of changes in these variables upon the overall system performance.

3. Functional routines of the simulator

This section will give a general description of each of the functional routines within the simulator. Figure 1. is a block diagram of

these functional routines.

A. Start New Product

The time cycle begins at this section and continues through the END-OF-WEEK HOUSEKEEPING section. The product, item and branch data cards are read into the program at this point. All the parameters of the system have now been set and the simulation of a product can now commence.

B. Initial Forecast of Demands

The twelve monthly demands for each item-location are averaged to provide a mean weekly demand rate. There is some correlation introduced here between the predicted demands and the actual demands that the branches will experience later since both the initial predicted demands and the actual demands are based on the monthly demands. However, it is doubted that this correlation is sufficient to appreciably influence the system performance except under rather erratic demand patterns.

C. Initialize Order Queues

The Order Queues are initialized to ensure that any data left in the queues from the previous product simulation is not erroneously introduced into the present product simulation.

D. Compute Optimal Base Rules

The computation of the optimal base rules is really the heart of the simulation program. The decision rules or policies developed here will in effect make or break the system. In the present form of the simulator the optimal requisition objectives and reorder levels are those published in [1.]. The subroutine "RXCOMP" computes these levels based

on the initial mean demand rates of the item-locations. These levels are computed first in item units and then converted to weeks of supply. In the case of the central warehouse, its optimal requisition objective and reorder level is given as so many weeks of total item demand (all branches). Likewise, the system item reorder level is given as so many weeks of total item demand. The optimal requisition objective for all branches other than the central warehouse is given as so many weeks of demand at the individual branch. The product requisition level is taken as the item-location requisition objectives summed over all items and all locations. The optimal or economic order quantity (EOQ) is expressed as the ratio of the initial economic order quantity to the square root of the total product demand rate. (See figure 2. for a block diagram of this subroutine).

By expressing the initial requisition objectives, reorder levels and economic order quantities in these ways, they can be corrected or updated with the subsequent demand experience. For example, by expressing the requisition objectives for the central warehouse in weeks of total item demand, the corresponding number of physical units to use for the "current" week's requisition objective will be the "optimal" coverage period times the predicted item demand rate for the current week. Hence, as the predicted demand rate changes so will the requisition objective. The same principle applies to all the requisition objectives and reorder levels. In updating the economic order quantity the square root of the product demand rate is used since the demand rate enters the Wilson's EOQ[1] formula as the square root.

[1] Wilson, R. H., A New Method of Stock Control, Harvard Business Review, Vol. 5, 1926-1927, pp. 197-205.

E. Predict Demand and Generate Actual Demands

The subroutine PREDICT is used to forecast the weekly demand rates for each item-location, each item, and each product. The forecasting scheme is a nine week moving average. There is no trend correction (or smoothing) applied to the forecast. The item demand rate is the sum of the direct demands (does not include branch interim orders on the central warehouse) over all locations and the product rate is the sum of all item rates converted to product units.

Because the input data is expressed in months a complicated indexing system is necessary to select the most recent "nine week period" on which to make the forecast. (See the detailed description of this selection process given in the card write up section for card 137).

The subroutine GENDEM scales the monthly demand data down to a weekly rate. The indexing used here to select the appropriate month for scaling approximates the calendar month-to-week relationship as close as possible. (See the detailed description of this subroutine given in the card write up section for cards 913-939).

Figure 3. is a block diagram showing the relationship of these subroutines to the main program.

F. Compute R and X Levels

At the beginning of each time period in the simulation of a product, the requisition objectives and reorder levels are updated in accordance with the demands predicted for the period. Hence, these policy levels, as expressed in physical units, are changing each week but the "optimal" period of coverage remains constant. Under the SRI rules these "optimal" coverage factors are the outputs of the RXCOMP subroutine while under the alternate rules the coverage periods are given as the branch and plant

(or central warehouse) goals. It should be noted that under the later rules the reorder levels are taken as a percentage of the requisition objective. This percentage can be arbitrarily set by changing the value of the variable RGX on card 31.

G. SRI Production Ordering Procedure

The decision whether or not to place an order for new production is made on the basis of the system-wide inventory position of the product and/or of the item, and the inventory position of the item at the Central Warehouse. Each product and each item within a product classification has an assigned "optimal" system-wide reorder level. In addition, the central warehouse has an assigned reorder level for each item in its local inventory. Each branch also has an assigned reorder level but the fact that a branch (other than the central warehouse) has fallen below its item reorder level will not trigger an order for new production. It will, however, trigger an order for an interim branch requisition on the central warehouse stock. When any one of the three inventory positions (i.e., the system-wide product level, the system-wide item level, or the central warehouse item level) falls below its reorder level a decision is made to place an order for new production. The system-wide inventory position of the item is reviewed to prevent imbalances between items of the same product since as mentioned before no provision is made to substitute one item for another. The item inventory position at the central warehouse is checked to insure that any imbalance between branches (with respect to the same item) does not cause eventual shortages at the branches with the deficiency. Since there is no redistribution between branches, the central warehouse must maintain its stock levels even though there might be sufficient stock of the product and/or the item in the system.

Once the decision is made to place a production order it becomes necessary to determine what quantity to produce. Since the underlying philosophy of the supply system is to anticipate branch needs as accurately as possible and to automatically resupply them with as few interim shipments as possible the production allocations are based on the latest possible inventory position. Hence a deferred allocation procedure is utilized. The initial production order quantity is taken as the difference between the branch requisition objective and inventory position levels summed over all branches and all related items of a product in product-units. The quantity is then adjusted to the next higher production batch size (you might manufacture 10,100 yards of broadcloth, but not 10,001).

H. Common Production Ordering Section

After determining the production order quantity of the product an initial (or tentative) allocation is made to the related items and an item-quantity allocation is also made to the central warehouse. These allocations are made to record the "dues from production" on the central warehouse records and the system-wide product and item records. This is necessary in order to prevent production orders from being placed again at the next review based on the same needs that trigger the current production order. It should be noted that item-quantity allocations and central warehouse allocations are made to the nearest packaged lot (e.g., case lots). The production order quantity, delivery date, allocations to item-quantities, and allocations to the central warehouse are then stored in a queue to simulate the production-run leadtime delay. Production order "dues" are not established at the branches but only for the central warehouse and the system-wide item and product records. This is equivalent to assuming that production order allocations are received at

the branches prior to making the allocations for the next production run.

I. Fill Production Orders

When the delivery date for a new production order comes due, the initial allocations to item-quantities and to the central warehouse are canceled and the system item needs are recomputed based on the current inventory positions of the branches. At this point in time the production-run component of the production order leadtime has already expired and only that fraction due to the waiting and shipping components remain. Since the requisition objective was determined as a function of the total lead-time, the branches would be over supplied if the deferred allocations were based on the optimal requisition quantity. To avoid this situation, the requisition objective is adjusted downward by the amount of the average demands during a production-run leadtime. In determining the needs at the central warehouse both its direct demands and branch reorder demands are considered. Since the quantity of the basic product produced may not be the same as the sum of the current item needs the item allocations are taken as the current needs times a proportionality factor which is the ratio of the quantity produced to the sum of the current item needs. In the simulator this quantity is then rounded to the nearest packaged lot (case). These quantities are then delivered to the central warehouse whose records are updated. Immediately afterwards, allocations are made to each item-location based on the adjusted requisition objectives and current inventory positions and placed into the branch order queue to await delivery at the appropriate time (see figure 5). The allocation to the warehouse is in effect the residual quantity not required to fill the branch needs. It is tacitly assumed in the simulator that the

quantity in stock in the warehouse plus the quantity needs of the warehouse are sufficient to cover any increase in total branch needs due to scaling and rounding. This assumption seems valid except in the instance the warehouse does not stock the item and has had no demand and even in this case the probability of exceeding the production quantity to any significant extend would seem very small.

J. Alternate Production Ordering Procedure

Under this procedure production orders are placed at each scheduled review period. Ths production order quantity is based solely on the needs of the central warehouse since under this concept all branch needs are filled by placing orders on the central warehouse. No direct allocations are made to the branches. Hence, this system can be thought of as a "pull" system. A minimum production order quantity is set to ensure economic plant operations.

K. Fill Branch Orders

This section simulates the delivery of both branch interim orders and production allocations at the branches. The branch order queue is screened to see if the delivery date of any order is due and, if so, the on-hand, on-order, and in-transit records are updated accordingly. Figure 4. is a block diagram of this section of the program.

L. Fill Demands on Branches

In this section the actual weekly demands are presented to the branches for issue. If the on-hand balance is not sufficient to cover the weekly demand, stock may be issued in partial fulfillment of the request or the demands may be canceled. If the stock out policy selector (ICON3) is set at 1, the requests are partially filled, if it is set at

2, the request is canceled. If the request is only partially satisfied the balance is canceled since backorders are not considered in the present form of the simulator. The tallies representing the item-location on-hand balance, inventory position, satisfied demands, and total demands received to date are updated as well as the system, item and product inventory positions.

M. SRI Interim Branch Ordering

Although the system is designed to resupply the branches by allocations from the production orders in sufficient time to meet their expected needs, unusual demands may reduce a branch's item level sufficiently to warrant an interim resupply order. Consequently at each review time the item-location inventory positions are screened against their reorder points. If the inventory position is at or below this level, a decision is made to resupply the branch with the particular item(s) from the central warehouse.

After making the decision to place an interim branch order we must determine the quantity to requisition. As in determining the branch need for the deferred allocation we must adjust the optimal requisition level downward by the average demands during the period between the placing of the most recently delivered production order and the present time. In so doing we are in effect compensating for only those demands which are above the normal rate and hence, are allowing the system procedures to function with the least interference. The difference between this adjusted requisition level and the current inventory position is the "optimal" interim order quantity. However, in some instances, depending on the relative lengths of a production order cycle and the production leadtime, this quantity may be too low and a minimum quantity must be set.

In determining the minimum value of the interim order quantity when the production leadtime is short compared to the production order cycle, it may be assumed that the need for reordering will occur within a production leadtime, (i.e.) after an order for new production has been placed but prior to its receipt at the branch. It may also be assumed that this need will occur within the latter half of the production leadtime. Hence, if the branch inventory position is raised to the reorder level plus one half of the expected demands during a production leadtime this quantity should suffice until the next production allocation is received.

On the other hand, if the production leadtime is long compared to a production cycle (period between placing orders for same product) the deliveries will be more frequent with perhaps smaller quantities and the need to order will occur most likely within a production cycle and presumably within the last half of that cycle if the system is performing fairly close to expectations. Hence, in this case the minimum interim order quantity should be the difference between the reorder level plus the expected demands in one half a production cycle less the actual inventory position.

The actual interim branch order is then the maximum of the "optimal" quantity and the smaller of the minimum order quantities. In the simulator this order quantity is then taken as the next highest case or pallet lot depending upon whether the additional holding costs are greater than the additional handling cost incident to processing broken pallet loads.

Once the order quantity is finalized, the simulator updates the warehouse records and stores the order with its delivery date in the Branch Order Queue (see figure 5.). At the appropriate time the order

quantity is removed from the queue and the branch records are updated accordingly.

N. Alternate Branch Ordering Procedure

Since branch orders are only placed at the scheduled review times the simulator first checks to see if a review will be made in this time frame. When a review is to be conducted, the stock status of the central warehouse is first checked to ensure availability of stock. If the central warehouse inventory position is below its reorder point the requirement at each branch is cut in half to supply some stock to all the branches which require replenishment. It is implicitly assumed in both instances that the total quantity supplied to the branches does not exceed the warehouse's on-hand balance. It should be noted that there is no provision made for placing branch orders at times other than at the scheduled review date. For example, if a zero balance is reached no action is taken until the next review regardless of how long that interval might be. If the review interval is one time frame unit there are no difficulties. However, if the review period was several time frames in length and if performance of the system under the alternate rules were to be compared to its performance under the SRI or any other set of rules this limitation might be significant.

O. End-of-Week Housekeeping

A miscellaneous section where the average inventory on hand, the average value of the inventory in-transit and the number of weeks of stockout tallies are updated. There is also an optional print statement to produce an inventory status report for each item-location for each time period. This section also marks the end of a time frame (week).

The program either returns to the predict section to begin the next time frame or continues on to print the output reports if the present product simulation is completed.

The program either returns to the report section to obtain the next line items or continues on to print the action reports if the present Product simulation is completed.

SCHEMATIC DIAGRAM OF SIMULATOR

PROGRAM INVSIM

START NEW PRODUCT

Read input cards

INITIAL FORECAST

Compute mean demand rate over last twelve months

INITIALIZE ORDER QUEUES

Zeroes out all the locations reserved for the production orders and branch orders in the SUBROUTINES QUEUE P and QUEUE B

COMPUTE OPTIMAL DECISION RULES

Call subroutine RXCOMP to compute optimal coverage periods

GENERATE ACTUAL DEMANDS

Scale monthly demands to weekly rate

PREDICT DEMANDS

Forecast demands for the period using a nine week moving average

COMPUTE R AND X LEVELS

Update optimal rules with current demand rates

Figure 1

FILL PRODUCTION ORDERS

Simulates delivery of the item from the plant to the warehouse and makes final allocations to items and to branches. Places allocations in the branch order queue.

FILL BRANCH ORDERS

Simulates the receiving of both interim orders and production allocations at the branches

FILL DEMAND ON BRANCHES

Simulates issues or sales of items at the individual branches and updates inventory records

SRI INTERIM BRANCH ORDERING

Determines whether a branch needs an interim replenishment and if so the appropriate quantity to order from the warehouse

SRI PRODUCTION ORDERING

Decides whether or not to order and computes order quantity

COMMON PRODUCTION ORDERING

Makes initial allocations of the order quantity to the central warehouse and to items for record purposes. Places order in the production order queue

END OF WEEK HOUSEKEEPING

Miscellaneous section where certain inventory tallies are updated

Figure 1

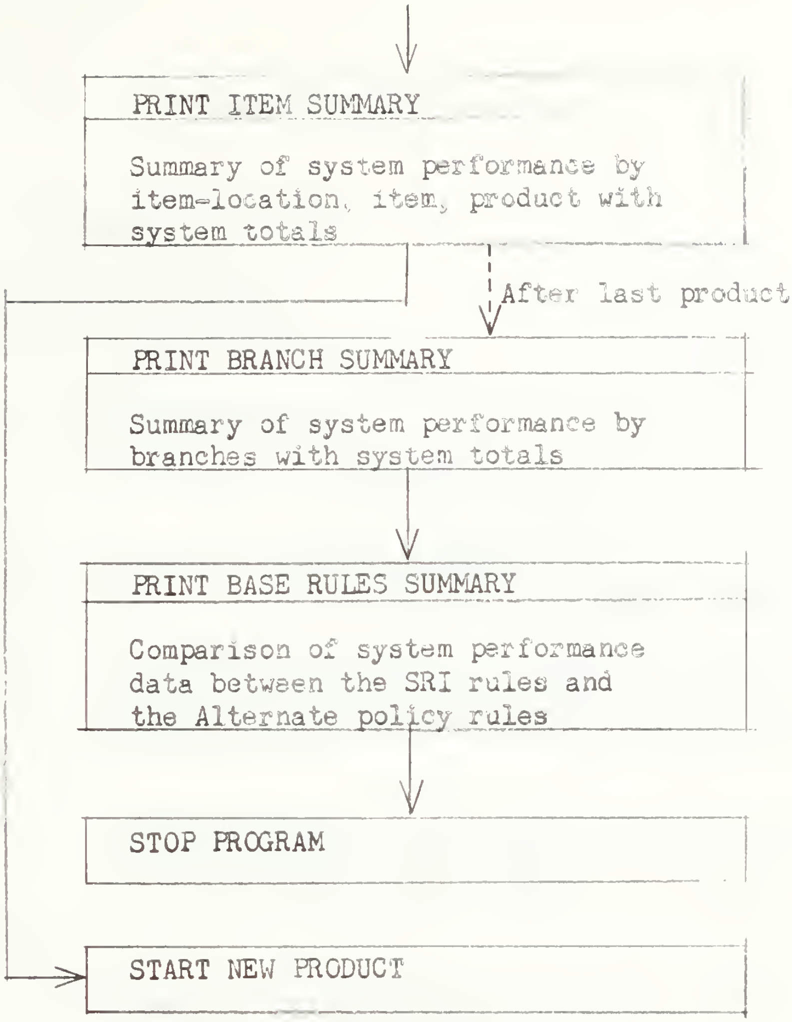
PRINT ITEM SUMMARY
Summary of system performance by item-location, item, product with system totals
After last product
PRINT BRANCH SUMMARY
Summary of system performance by branches with system totals
PRINT BASE RULES SUMMARY
Comparison of system performance data between the SRI rules and the Alternate policy rules
STOP PROGRAM
START NEW PRODUCT

Figure 1

COMPUTE OPTIMAL BASE RULES

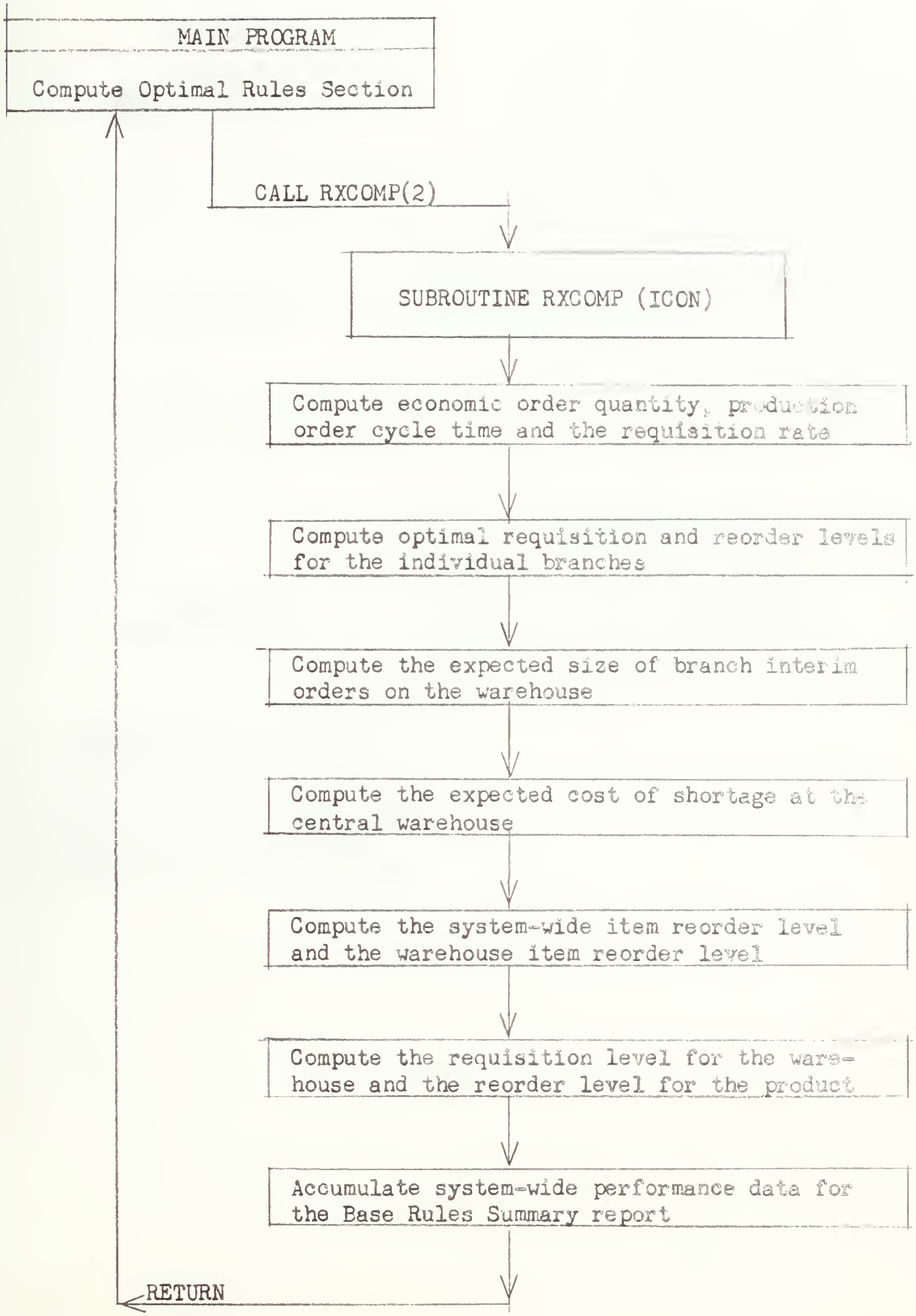

Figure 2

PREDICT DEMANDS AND GENERATE ACTUAL DEMANDS

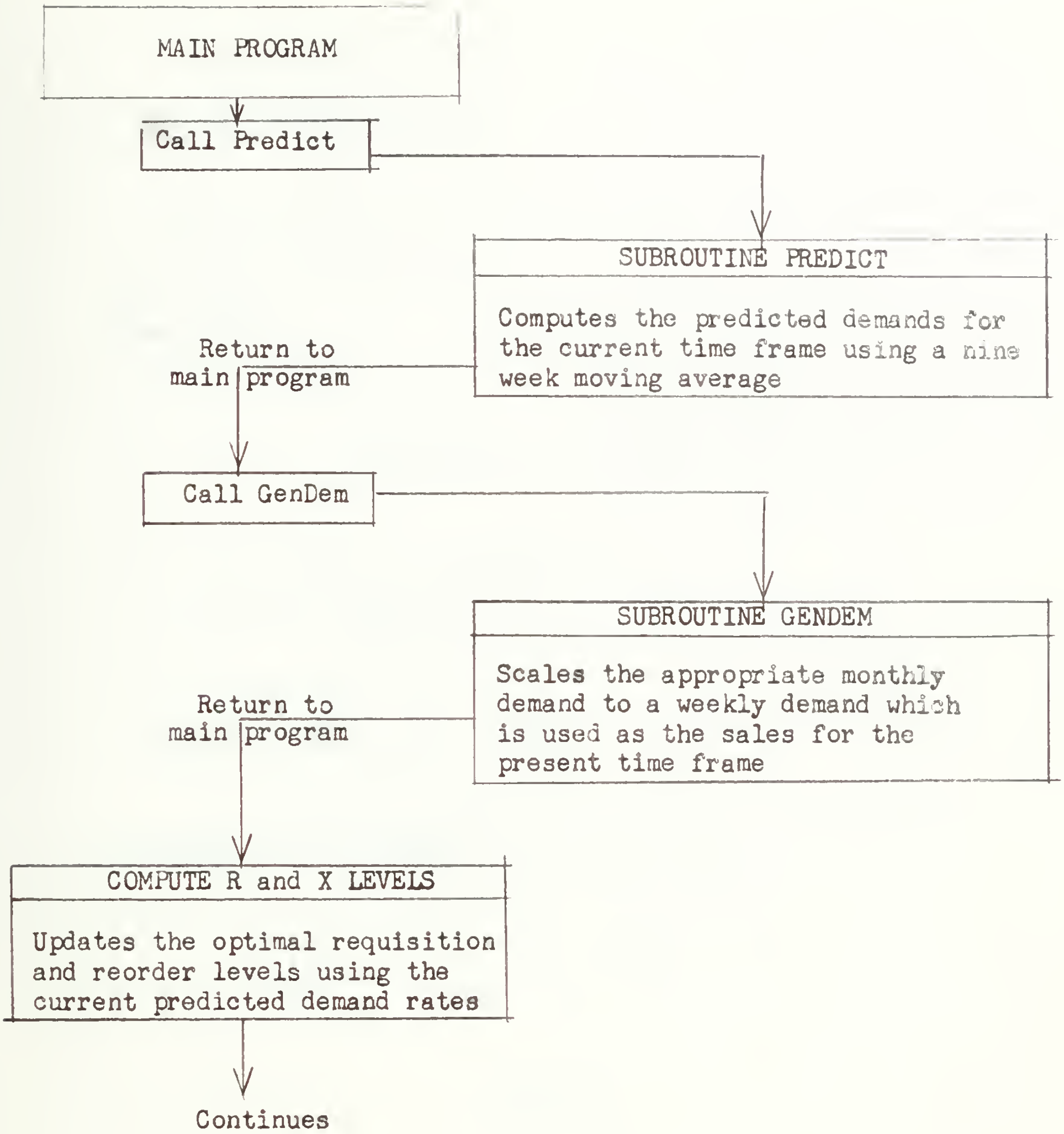

Figure 3.

RECEIPT OF ORDERS AT BRANCHES

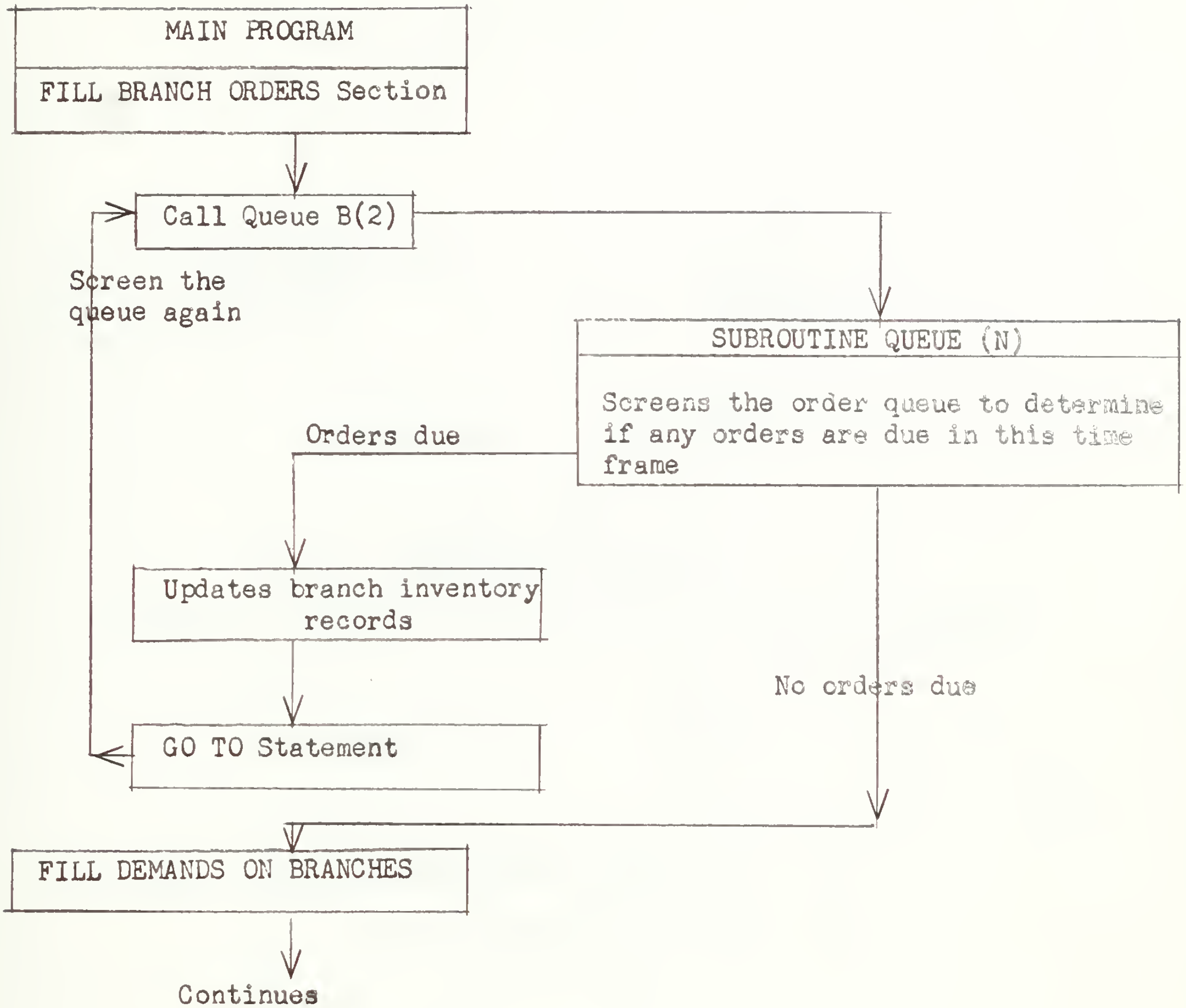

Figure 4

PLACE ORDERS IN BRANCH ORDER QUEUE

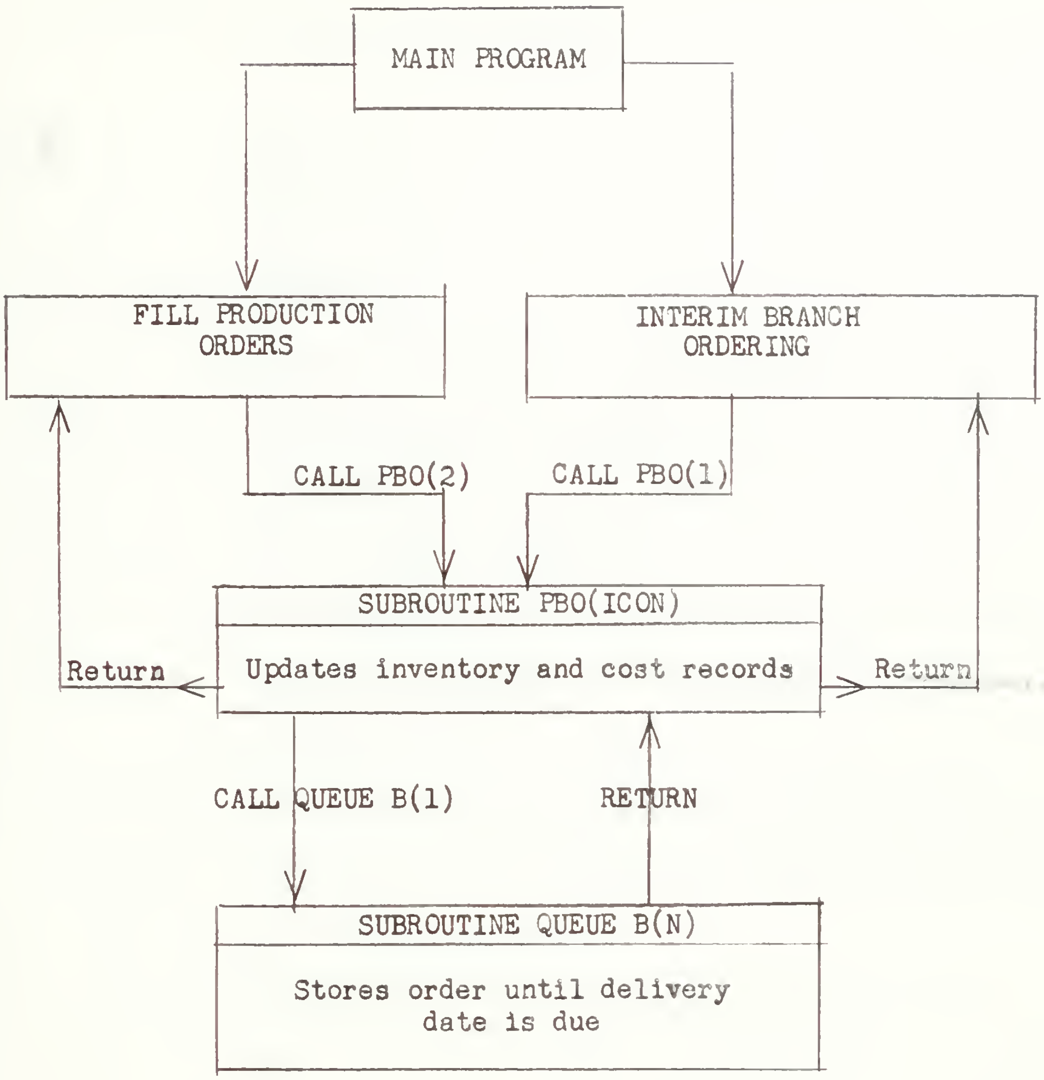

Figure 5.

CHAPTER III

DETAIL DESCRIPTION OF THE FORTRAN LANGUAGE CODING OF THE PROGRAM

Card No.	Description
1	Program Name
2-4	COMMENT cards
5-8	Dimension Statement. These variables are used in main program in addition to those variables that are listed in common. There are 25 possible branches and 20 possible items or sizes.
9-27	Common Storage. Dimensioning and placing of variables used in main program and subroutines into common storage locations.
28-37	Start of Program. Initialize various variables and setting End of File flag to 1. Program stops on End of File flag set to 2.
38-41	Read in the Control, Base Rules parameters, and Branch data cards. These decision rules and parameters are now set for the entire run. If different decision rules and/or parameters are to be used new runs will have to be made separately. The variable Z is used as a dummy for both the obsolescent cost and the variable VLP1 which is no longer used.
42	The opportunity cost is read-in in the same format as the physical storage cost per unit (APB and APP) and must be multiplied by .1 to obtain correct value. This is done solely for convenience of formatting.

43-47 Control Data, Decision Rules parameters and Branch data are printed out to check correctness of read-in and also to have this information listed on the out-put sheets for future identification and reference.

48-50 Leadtime components are read-in in "days" and are converted here to "weeks".

51 Call SUBROUTINE RXCOMP to initialize variables used for system totals.

52 Call SUBROUTINE RDTP to read-in the first product characteristics, i.e., the base number assigned to the product; the shelf life expressed in months; the branch goal expressed in days; the plant goal (central whse. stock) expressed in days; the production lead-time expressed in days; the batch size increment ex-pressed in product units; minimum batch size expressed in product units. All other product cards are read by card number 105.

53-54 Use a do-loop to set all of the "intransit" leadtimes for the branches to same value. This is equivalent to assuming branches are equidistant for shipping purposes from the main warehouse/plant.

55 Re-set the warehouse shipping leadtime to zero. Since the warehouse is located at the manufacturing plant there is no shipping time involved.

56-68 Use a do-loop to initialize branch variables.

69-72 Start New Product: Set the time clock to zero and initialize the product variables.

73-79 Convert shelf life, branch goal, plant goal, and production leadtime to "weeks", and transfer the values of batch size increment (BSI), batch size mean (BSM), and base number (BASE) to permanent storage locations. It should be noted that all product, item, and branch characteristic data are first read into the AO array and then transferred to permanent variable name locations. When the program was used by SRI this data was read-in from a tape rather than cards and having a standard format for presenting information simplified the problem of handling massive amounts of data.

80 Call the subroutine RDTF for the first item of the present product. Here the size code, item number, cartons per pallet, units per cartons, weight per unit, production unit cost, selling price, and size conversion factor (# of product units per item unit) are read. All subsequent item, and branch data cards related to the present product are read by the "Call RDTP" on card number 105.

81-82 Set the count for the different items (sizes). As each item card is read the program returns to card 82 where the item count is increased by one.

83 Initialize the inventory position of the item.

84-95 Initialize the branches' (item-location) inventory records; satisfied demands, order count, stock out weeks, stock out counts, broken case count, broken

pallet count, on order quantity, average inventory position, total demands and the branch flag.

96-104 The product number (four digits) and the size code (two digits) are listed as a six digit number on the data card. Cards 96, 97, and 98 are used to separate these two numbers and store them in permanent location and to scale data as necessary.

105-124 On the first pass through this section of the program we have already read the first product card and the first subsidary item card. When we call "RDTP" the first time at card 105 we will read a branch data card which will leave a value of 3 in the AO(1) location. By testing this variable, the next two cards will send us to card 110 where we select out the proper value of k by matching the branch code array to the branch code number read from the data card. Having now determined the proper item and branch numbers (j and k) we transfer the input data to permanent storage arrays. As a bonus operation we also compute the mean item-location demand rate and the item and product inventory positions.

125 This card sents us back to card 105 where we call RDTP again and read the next card. Four sequences are possible at this juncture:

(1) If the next card is another branch usage data card we will repeat the same sequence as outlined above.

(2) If the next card read is an item record card, AO(1) will be set to 2.0 and this time card 107 sends us back up to card 82. In this way the machine recognizes that no more branches are carrying the previous item and that a new item is now being considered (but still under the same product). The item count, NJ, is increased by 1 and we initialize all the item and branch variables for the second item. We are then back to card 105 where we call RDTP in search of branch usage data cards for the new item (size). We now follow sequence (1) again.

(3) If a product card is read, AO(1) will be set to 1.0 and card 107 will send us to card 129 where we commence simulating for the previous product and subsidary items. Hence, the data for the next product remains in the AO array until the simulation is completed for the previous product, at which time the program returns to card 69.

(4) If, when card 125 sends us to card 105, a blank data card is read, the RDTP subroutine sets AO(1) to -1.0 before returning to the main program. Hence, when we test AO(1) this time we are sent to card 108, which

sets End of File to 2. We are then sent to card 129. We have completed reading in all the data cards and will simulate for the last product, print out the summary statements and stop the program.

126-130 Initialize the Order Queues. Before beginning the simulation we wish to initialize the Production Order Queue and the Branch Order Queue. The present routine sets the initial values to 0.0 so in effect we are simulating a system from its beginning. If this is not desirable the program could be modified so that representative values could be read in to simulate a system in progress.

131 Compute R and X Levels. Here we check to see what set of Decision rules will be used in controlling the production ordering and branch shipments. If ICON1 has a value of 1 we use the SRI rules, otherwise we use the alternate set of policy rules.

132 The "RXCOMP" subroutine is called to obtain the optimal policy rules for the present product. The requisition objectives and reorder levels for each item location, the system-wide item and product reorder levels and the minimum economic order quantity are computed within this subroutine. The requisition and reorder levels are expressed in weeks of supply rather than in item or product units so that they may be varied as the current demands varies.

133-135 "Generate Actual Demand, Predict Demand, and Compute X and R Levels" heading.

136 Start clock by setting time equal to 1.

137 Call SUBROUTINE PREDICT to make forecast of the weekly demand rate for each item-location, each item, and for the product. Demands are computed by taking a weighted average of the demands in those months in which the previous nine weeks occurred. For example, the prediction for the first week is based on the demands occurring in the nine weeks (the previous two months) immediately preceding the simulation period. It is assumed that five weeks occurred in one of the months and four weeks in the other. Hence, the demands in the past two months are weighted by five-ninths and four-ninths. This weighted monthly average is then scaled by 7/30 to obtain a weekly average. In the Simulator, the demands of the eleventh and twelveth months are also used as the demands for the two months prior to the simulation period.

In order to select the appropriate months in which the past nine weeks occurred, a complicated indexing system is needed. The schedule below indicates the results of this system for the first three time frames. The J1 index provides a means of assigning the nine week period to the appropriate months as the time index moves through the 52 simulation periods.

The DM(i) is the demands in the i-th month.

J1 Index	Monthly Demand			
45	DM(11)			
46	DM(11)			
47	DM(11)			
48	DM(11)			
49	DM(12)	T_1		
50	DM(12)		T_2	
51	DM(12)			T_3
52	DM(12)			
53	DM(12)			
54	DM(1)			
55	DM(1)			

etc.

138 Call SUBROUTINE GENDEM to generate the actual weekly demands by scaling the monthly demands inputted from the data cards by 7/30 and rounding to the nearest whole unit. Note that the predicted demands are designed as DIJK whereas actual demands are designated as DWIJK.

139 Branch to SRI rules or the alternate set as determined by the value of the inventory rule selector (ICON1).

140 Compute the economic order quantity (EOQ) for the current week. The variable FLQI is the ratio of the optimal economic order quantity to the square root of the mean product demand rate. By multiplying this

ratio by the current predicted product demand rate we are varying the EOQ to reflect the present demand trends. The variable FLDI is the maximum EOQ due to shelf life limitations.

141 Compute the production order cycle as the EOQ coverage period plus one half of the review interval.

142-143 Initialize the product requisition and set up a do-loop on items.

144-145 Compute the requisition levels for the central warehouse. The variable XRIJZ represents the optimal coverage period (of total item demands) and is converted to physical units by multiplying by the current predicted demands for the item. The variable XDIJK represents the maximum level due to shelf life limitations. The minimum level is given as the reorder point plus the demands in a production order cycle and leadtime.

146 The reorder level for the central warehouse is taken as the total demands (including interim orders) during a production leadtime. The variable XXIJZ is the ratio of the mean total demand during a leadtime to the mean direct demand rate.

147 The system-wide item reorder level is also taken as the demands during a leadtime.

148-150 Initialize the item requisition variable, establish a do-loop on locations and screen the branches to see if the item is carried.

151	The item-location reorder level is taken as the demands in a branch interim order leadtime.
152-153	The optimal requisition objective for the individual branches are computed by a formula analogous to that used for the central warehouse. However, the predicted weekly demands at the branch and its production order leadtime are used rather than the total item demand rate. (See cards 144 and 145)
154	Sum the requisitions objectives for each item over all branches except the central warehouse.
155	Continue statement for do-loop on locations.
155A-156	Compute the total requisition level for each item and each product.
157	The product reorder level is the product requisition level less the economic order quantity.
158-159	The variable FLPDTM represents a specific time. It is the time at which the most recently delivered production order was placed. The difference between this time and the present time represents that portion of the resupply cycle which has already expired. This period is used to adjust the branch requisition level when computing the order quantity for interim shipments. (See cards 250, 251)
160	Jump statement to skip over the following section which is used for the alternate policy rules.
161-167	The requisition objectives and reorder points are calculated for the central warehouse and branches

under the alternate policy rules. The warehouse levels are a function of the total item demand and a predetermined goal. The branch levels are functions of the local demand rate for a given item.

168-170 FILL PRODUCTION ORDERS Heading

171-172 Screen the production order queue to determine if any orders have been completed by the plant. If there is a positive order value due at this time, the program will continue on to simulate receiving the order in the central warehouse and making allocations to the individual branches.

173 Check to see which set of policy rules are in use. Under the SRI rules the program will compute the current needs of the branches and make automatic shipments to satisfy these needs. Under the alternate rules all production orders are delivered to the warehouse for storage. The branches place their orders on the warehouse as their reorder points are reached.

174-179 When production orders are initiated a temporary allocation is made for each item of the product based on the system needs at that time, and the system item and product records as well as the central warehouse's records are adjusted for the production "dues". These initial "dues" are cancelled here prior to making the actual allocations to items.

180 Compute the warehouse needs for each item of the product.

181-184 Compute the branch needs for each item and sum over all branch needs to give the total allocation of the product to each item.

185-186 Compute the total system current needs in product units and ensure that it is a positive quantity.

187-189 The item needs are scaled to the actual quantity produced (POBS) and rounded to the nearest case lot.

190-196 The appropriate inventory records are updated to reflect the delivery of the new items into the central warehouse.

197 Check to see which set of control rules are in use.

198-204 Compute the order quantity and delivery date for each item-location other than the central warehouse. The order quantity is rounded to a full case lot. The value of inventory in transit is also updated here.

205 The PBO (Place Branch Orders) subroutine is called to update the inventory records and cost tallies for each item-location. The parameter value of 2 in the Call indicates that this is a production allocation and not an interim branch order. The PBO subroutine then calls the Queue B subroutine and places the orders in the queue to simulate the shipping delay.

206-207 After completing the first production order the program returns to card 171 to screen the queue for any other orders that might be due.

208-210 FILL BRANCH ORDERS Heading

211-215 Screen the branch order queue to determine if any orders are due for receipt at the branch. If an order delivery date is due the on-hand, on order and in-transit inventory records are adjusted accordingly.

216 This GO TO statement cycles the program back to the CALL QUEUE B card to have another look at the queue.

217-219 FILL DEMANDS ON BRANCHES heading

220-221 Set up do-loops on items and locations.

222 Check to see if Branch carries this item. If not, machine jumps to the next K value and checks again.

223-224 Check to see if there has been any demands this week for the item-location. If so, the total demands to date tally is increased by the current demands.

225-241 If the on hand balance is positive we compute the new on hand balance (TP) and the quantity of demands satisfied. If the new on hand is also positive we jump to statement 523 to update the branch, item and product inventory records. If the new on hand balance TP had been negative or zero the program checks to see which issue policy is in use. If issues are to be made to a zero balance (a setting of 1) the old on hand balance becomes the satisfied demand (TP2) and the new on hand balance is zero. In this case the stockout count tally is also increased by 1 since only partial issue could be made. If on the other hand the policy is not to make partial issues the old on hand

balance is placed in TP and the satisfied demand (TP2) variable is set to zero and stock records are updated accordingly.

242 Branch to the SRI rules or alternate policy rules.

243-245 SRI BRANCH ORDERING PROCEDURE heading

246-248 Set up do-loops on items and locations and test to see which branches carries the item.

249 Review the inventory position to determine if a branch interim order should be placed. If the inventory position is at or below the reorder level an order will be placed.

250 Compute the time period between the present time and the time at which the most recently received production order was placed.

251-252 When placing a branch interim reorder the branch inventory position is brought up to its optimal requisition level. To do so would overstock the branch and interfere appreciably with the automatic production allocation system. The policy is to supply the branch with enough stock so that at the time just prior to receipt of the next production allocation the branch will be at its reorder level. Hence, the optimal requisition level is adjusted downward by the expected demands during the period discussed on card 249. A minimum reorder quantity is taken as the reorder level plus the demands in half a production leadtime (or cycle whichever is less) minus the

current inventory position.

253-254 Check to see if rules call for rounding order quantity to the nearest carton and round if they do.

255 Check to see if the rules call for rounding to pallets.

256-257 Compute cost of issuing broken pallet load for this order. The cost is the number of units in the broken load times the cost of handling such units (CBPC).

258-259 Compute the extra holding costs that will be incurred at branch if a full pallet load is issued.

260-261 If the holding cost is greater or equal to the broken pallet handling cost then the order quantity is taken in full case lots. If not, the order quantity is taken as the next higher full pallet quantity.

262 Compute delivery date for the order. Delivery date is specified as the time that the last scheduled delivery was made from central warehouse plus the scheduled waiting period plus the in-transit time.

263 Update the value of the inventory in-transit.

264 Call subroutine Place Branch Order to simulate the updating of the branch and central warehouse inventory positions, the on order quantity, the records for the branch, the on hand balance for the central warehouse, and the branch cost records. Subroutine Queue B is called from within the PBO subroutine to place the order and delivery date in the Queue until the appropriate time to simulate receiving the order at the branch.

265	CONTINUE Statement
266-268	SRI PRODUCTION ORDERING PROCEDURE heading
269-284	Check Product Inventory position against product trigger level. If inventory is above trigger level then central warehouse inventory is checked against its trigger level. If it is above the trigger level, then the item inventory position is checked against its reorder level. If all three are above their trigger levels no production order is placed. However, if anyone of the three is below its trigger level the production order quantity is computed by summing the difference between the requisition objective and the present inventory position over all branches and all items, rounded to the nearest batch size increment. The individual item needs for the system POAS(J) and the central warehouse POAW(J) are retained in order to make the necessary "dues" on the item and central warehouse inventory records. This eliminates the placing of another production order later on to fill the same needs.
286-287	The production order quantity is rounded to the next higher batch increment. The program now jumps to card 319 to complete the ordering routine. The intermediate sections pertain to the alternate policy rules.
288-318	(See Alternate Branch Ordering and Production Ordering Procedures at the end of this section).
319-327	Compute the scaling factor due to rounding to nearest

batch size increment (BSI) and the order delivery date. Production order allocations to size and to the warehouse are scaled to the production quantity; the warehouse's on order and inventory position, as well as the product and item inventory positions, are updated.

328-329 A tally is kept on the number of production runs made during the simulation. Finally, the order quantity and delivery date are placed in the queue to simulate the production delay.

330-332 END-OF-WEEK HOUSEKEEPING heading

333-339 Set up do-loops on items and locations and select out those locations which stock the item. The average inventory on hand and stock out weeks records are updated.

340-345 This is an optional PRINT statement used mainly for checking out the simulator. It provides a weekly status report for each item-location showing the on hand balance, the on order quantity and the inventory position. It also provides the system inventory position for the product and each related item.

346 Average Value of inventory in transit is computed.

347 Check the time to see if the simulation for this product is complete. If it is, the program will print out the item and product summaries. If not, it will return to the Generate Demand Section, set the time to the next increment and go through

program again.

348-350 Print Routine for Item Summary.

351 Update the page number tally.

352-358 Set up do-loops and select out Branches carrying each item and print item-location data:

DWIJK	Actual weekly demands in units
RIJK	Requisition level in units
XIJK	Trigger (reorder) level in units
XRIJK	Requisition level in weeks
XDIJK	Demands in a shelf life, item-location
XDIJ	Ratio of total warehouse demands for an item to total system direct demands for that item
XXIJ	System item reorder level in weeks
XRIJZ	Warehouse item requisition level in weeks of total system direct demands for that item
XXRIJZ	Warehouse item reorder level in weeks of total system direct demands for that item
XK2IK	Interim branch order leadtime
SIJK	Inventory position of item-location

359 Print page heading

360-386 Initialize product and item print variables prior to their computation by summarizing item-location data.

387-429 In this section the system performance data is summarized for each item with individual branch subtotals. The inventory values are given in both units and dollar figures. The effectiveness measures such

as the stock turn ratio and the ratio of lost demand to total demand are based on units rather than dollars. The holding cost, stockout costs, and costs of handling broken carton and pallet lots are also computed for the summary report. The print statement causes the item-location data to be printed.

430-445 The inventory data is summarized by product totals in this section. The print statement here causes the item totals to be printed on the summary report.

446-450 The product effectiveness measures (stock turn ratio, loss demands to total demands, and weeks per stockout) are computed. The print statement prints the product totals on the summary report.

451-463 The overall system totals (all products) are accumulated here for the Item-Product Summary Report.

464 Test to see if the simulation of all products is complete. If not, the program returns to the START NEW PRODUCT section and commences simulation of the next product.

465-467 PRINT BRANCH SUMMARY AND STOP heading

468-473 The overall system effectiveness measures are computed and the system totals are printed for the summary report.

474-486 Initialize the system variables prior to recomputing by summarizing on items and products for the branch totals.

487-510 Compute the system totals by summarizing the branch

totals. The individual branch totals are also printed within this do-loop.

511-513 Print the system totals.

514 The cost of all production set ups.

515 The total branch variable costs for the system.

516 Page number tally.

517 Print the page heading.

518-519 Branch on the control rules. If the SRI rules are being used, the program will print a Base Rules Summary which compares the system performance under the SRI rules with those of the alternate policy set.

520 Print branch variable cost factors and the value of the average inventory in transit.

521 Stop the program card.

522-581 Formats for READ and PRINT statements.

SUBROUTINE RXCOMP (ICON)

582 Subroutine defined with call parameter ICON.

583-588 Dimension the local variables.

589-606 Dimension and place in Common those variables used in one or more subroutines of the program.

607-610 Function VARDF (D,C) defined. See SRI T M -3 for information on equation used. It computes the rel-variance (ratio of the variance to the square of the mean) of the demand distribution.

611-612 Computed GO TO Statement for branching into one of the three subdivisions of the program. The first is merely

an initializing routine; the second is the computation of the base rules themselves; and the third is the recap and printing of management data taken from the base rules.

613-627	Routine for initializing system management variables.
628-631	COMPUTE BASE RULES heading
632-635	Initialize variables to be computed.
636	Computation of expected leadtime between reaching a trigger level and availability of item for distribution or the production leadtime for central warehouse. Sum of half of the review interval, the communication leadtime, and the production-run leadtime.
637	Set L. C. Up do-loop for locations
638	Compute the mean production order leadtime (reorder-to-receipt) for each branch.
639	Compute the mean branch interim order leadtime (reorder-to-receipt) for each branch. It does not include the communication leadtime or the production-run leadtime.
640	The item-location reorder point is taken as the demands during a branch interim order leadtime. Here the branch interim reorder leadtime (FLIK) is multiplied by the variable FK2. For management purposes, studies can be made of how the reorder level influences the overall performance of the system by changing the value of this leadtime through the multiplier FK2.
641	T5 is an intermediate computation in the variance of

leadtime. It is assumed that the times when production is actually needed are distributed uniformly between times of review, and that the times of availability are distributed uniformly between times of regular (scheduled) branch shipments. If the above assumption is true, the variance of the reaction component of leadtime is $FLRI^2/12$ and the variance of "waiting for shipment" time is $FSK(K)^2/12$.

642 The rel-variance (ratio of the variance to the square of the mean) of the production order leadtime is computed as the sum of the variances of the five independent components (reaction time, communication time, production time, waiting time, and shipping time) divided by the square of the mean leadtime, FLIK(K). The rel-variance of the production-run component is an input parameter and is used here to compute the variance of the production-run leadtime. The shipping times and the communication times are taken as nearly constant with zero variance.

643 T1 - Intermediate computational factor used to simplify calculations later.

644 Set up do-loop on items. This loop is inside the do-loop on locations.

645 Test to see if branch carries item.

646 Dealer or Branch unit cost of an item. Sum the production unit cost CJ(J), unit freight cost, and unit receiving cost.

647 Compute the rel-variance of the demand rate as a function of weekly demand rate and dealer's unit cost. (See [1.] pages 21 and 22 for reference.)

648 DIPR - Rate of arrivals of orders for the product (per unit time) computed as the reciprocal of the rel-variance of demand. (See [2.] page 25)

649-651 FIJK represents the demands during a production order leadtime adjusted for the variances of the demand and leadtime distributions. (See [1.] page 20). T2 is an intermediate factor in computing the variance multiplier (FLKIJK).

652 F is the sum of all the FIJK'S converted to product units. It is a factor in the computation of the economic order quantity for the product.

653 Unit holding cost per unit of time is taken as the sum of the opportunity cost and physical holding cost. (Cost of holding one unit of an item for one year at a given branch).

654 DICHI - Factor in computing the Wilson economic order quantity (EOQ). In order to determine the average unit holding cost (over all branches and items) a weighted average of branch holding cost is taken. The weighting factor is the ratio of the item-location demand rate to total product demand rate. The quantity DICHI is this weighted average times the total product demands rate.

655 Sum the predicted weekly demands by location to give

the predicted weekly demand rate for an item.

656 Sum the predicted weekly demand rate by items to give the product demand rate in the product-equivalent units. SJ(J) is the ratio of product units to item unit.

657 CONTINUE statement

658-660 This PRINT statement is used for checkout purposes only. The values of the print variables are checked against those computed manually to verify the accuracy of the program. These cards may be removed after the program is checked out.

661-662 Computation of Economic Order Quantity heading. The optimal economic order quantity is computed as a modified "Wilson" EOQ using the following formula:

$$Q = \begin{cases} Qw + f & \text{for } \left(\dfrac{Qw}{f}\right)^2 \geq 2.25 \\ 1.85(Qw^2 f)^{1/3} & \text{otherwise} \end{cases}$$

where

$Qw = \sqrt{2\ AD/C_h}$

A is set up cost

D is product demand rate

C_h is average holding cost per unit per year

f is demands during a branch production leadtime adjusted for variances.

663 Compute the square of f.

664 Compute the average holding cost over all branches.

665 Compute the square of the Wilson's EOQ divided by

f squared. The variables CM2 and FMU are multipliers and would normally be set to 1.0 and 0.0 respectively.

666 Test the value of SQI to determine which formula is appropriate.

667-668 If we are in the first region of the formula, we convert the test quantity, $(Qw/f)^2$, to the appropriate formula by taking the square root, adding one and multiplying through by f.

669-670 If the second formula is to be used then the square of the Qw/f ratio as computed on card 665 is multiplied by the constant 1.85 and raised to the one third power. This value is then multiplied by the factor f to obtain the correct formulation.

671-673 The maximum or upper limit on the EOQ is taken to be the demands during the shelf-life less 1.5 times the expected demands during the central warehouse's lead-time. The minimum of the optimal and maximum values is taken as the final EOQ for the product. This quantity is rounded to the next higher unit.

674 The discrete adjustment period is the expected reaction time (FLRI/2) plus the expected time between orders (one half of the reciprocal of the requisition rate).

675 The adjusted EOQ coverage time is taken as the production order cycle (the EOQ divided by the demand rate) plus the discrete adjustment period. This later period accounts for the fact that demands arrive

in "lumps" and also for the delay between reaching a reorder level and taking action to reorder.

676 QIPR is the number of orders corresponding to the adjusted EOQ quantity.

677-678 PRINT statement for checkout purposes only.

679 FLQI is the ratio of the optimal EOQ to the square root of the predicted demands for the product at the beginning of the simulation period. This ratio is multiplied by the square root of the current predicted demand at the beginning of each week in the main program to update the EOQ value.

680 FK4 and FK1 are the same multiplier. The first is used in the RXCOMP subroutine and the later is used in the main program. They are both used to vary the leadtime demands when computing the minimum requisition level. (See card 720 and 153)

681-685 Initialize the product variables.

686-693 Set up a do-loop on items and initialize the item variables which are to be computed by summing over locations within the do-loop on branches. Note that the central warehouse is included in the sum.

694 Test to see if the branch carries the item.

695 Compute the dealer's (branch) unit cost on an item which included the freight and receiving cost per unit in addition to the production unit cost.

696 Compute the rel-variance of the demand rate on an item location as a function of the predicted demand

rate and the branch unit cost.

697-700 Compute the demands during a production leadtime adjusted for the variances due to leadtime and demand distributions.

701 Compute the holding cost for each item at each location.

702 FKIJK is a variance factor used in computing the requisitioning level for each item-location.

703 The cost of a shortage at a branch is taken as the profit lost when out of stock.

704 The probability of ordering.

705-706 Compute the optimal requisition objective level for each item-location. (See Chapter II for details of the formula)

707 Any system and/or branch constraints are applied here. In utilizing these constraint factors the immediate effect is to reduce the requisition objective. This, of course, will result in a lower average inventory and lower holding costs. The system constraint factor may be used with or without any branch constraints.

708 Compare the optimal requisition level to demands during a shelf-life and take the minimum.

709-711 Print the item-location decision rules and input data for check out purposes only.

712 T5 is an intermediate factor in computing the rel-variance of leadtime. (See card 641)

713 Compute the rel-variance of the branch interim order leadtime.

714-717 Compute the demands during a branch interim order leadtime adjusted for the variances of the demand and leadtime distributions. It should be noticed that the factors computed here on cards 712-717 are analogous to those computed for the production order leadtime on cards 641-642 and 697-700. These variables are used in computing the warehouse's optimal requisition level.

718 CKZJK is an intermediate factor used in computing warehouse requisition level.

719 The branch reorder level is taken as the interim order leadtime demands.

720-721 The optimal requisition level is compared to a minimum value which is taken as the reorder point plus the demands in a production order cycle and leadtime.

722 Sum the item-location requisition levels over all branches other than the central warehouse.

723-725 DZIJK is the mean of a negative exponential distribution which is used to approximate the "tail" of the actual demand distribution. (See [1.], pages 29-31)

726 PRIJK is the probability that an interim order for the (J,K)th item-location will occur following the arrival of a production allocation and before the arrival of the next allocation. Or, the

$$P\left[D \text{ in } (t+FLIK) \geq RIJK - XIJK\right].$$

727-728 Compute the expected size of an interim branch order given that an order occurs.

729 DPR(J) is the expected size (number of units) of an interim order for the j-th item. This quantity represents that portion of the total demands on the central warehouse by interim orders from the branches.

730 Compute the inter arrival time of branch interim orders, i.e., the number of weeks per interim order.

731 Compute the probability of an order for the j-th item.

732 Compute the optimal requisition objective level expressed in weeks. (This card would more logically follow after card 722). In the main program, a new optimal requisition objective level is computed each weeks using this optimal coverage time. The new requisition objective level in item units is this coverage factor times the current predicted demands.

733 The weekly predicted demands are summed over all branches except the central warehouse.

734 The demands in a production leadtime are summed over all branches except the central warehouse.

735 T2 is an intermediate computation to be used in computing the average cost of a central warehouse shortage.

736-737 CSZ(J) is the average cost of a central warehouse shortage of the j-th item.

738	XDIJK(J,K) is the demand in a shelf-life period for the j-Kth item-location.
739-740	PRINT statement for checkout purposes only.
741	CONTINUE statement for the do-loop on locations.
742	Compute the total warehouse demand rate as the sum of the direct demand rate and the branch interim order demand rate. Prior to this statement DPR(J) was given as the number of units demanded in a production order cycle. Here this quantity is divided by the weeks per cycle (STI) to give the interim order demand rate.
743	Initialize the optimal requisition objective (expressed as weeks of supply).
744	XDIJK is the demand of an item over a shelf life period at the central warehouse. This quantity is used as the upper limit on the requisition objective.
745	XK2IK(1) is the reorder time for the central warehouse times a multiplier.
746	Compute the total direct demands per week for an item.
747	Compute the total direct demands in a production lead-time for an item.
748	Compute the total direct demands in a production lead-time for a product in product equivalent units.
749	Compute the total direct demands in a production lead-time for a product considering branches only.
750	Compute the total direct demands per week for a product considering branches only.
751	The system item-reorder level is computed as system

demands in a leadtime. The multiplier FK5 can be varied to increase or decrease this quantity.

752 Compute the total direct demands for all items of one product placed on the central warehouse.

753 XXIJ(J) is the system item reorder level expressed in weeks. In the main program this coverage period times the current predicted demand rate is used as the optimal reorder level for the current time period.

754 XDIJ(J) is the optimal reorder level for the central warehouse expressed in weeks of total demands for the item in the system. It is used in the main program to compute the optimal reorder level in physical units by multiplying it by the current item demand rate.

755 XJ1(J) is the central warehouse reorder level for the j-th item taken as demands in a production leadtime.

756 XXIJZ(J) is the central warehouse's reorder level expressed in weeks of total item demands in the system.

757 Sum the item requisition objectives to determine the product requisition objective.

758 CIJ1 is the unit cost of an item at the central warehouse; production cost plus delivery and receiving costs. The later costs are given in dollars per hundred weight and must be scaled down by .01 to give dollars per pound. The weight factor, WJ(J), is given in pounds per unit.

759 Transfer the unit cost figures to their permanent

storage locations.

760-761 Compute the rel-variance of the direct demands on the central warehouse, and transfer this quantity to a permanent storage location.

762 CHJJ(J) is the holding cost per unit per week. AZR is the opportunity or interest cost expressed in dollars per dollar of inventory per year and APP is the physical holding cost expressed in dollars per product unit per year.

763 VDPR(J) is the rel-variance of the total demand on the central warehouse for each item.

764 DIPRPR is the requisition rate corresponding to the total demands on the central warehouse for each item summed over all items to give requisition rate for the product.

765-766 PRINT statement for checkout purposes only.

767 CONTINUE statement for do-loop on items.

768 FK4 is a variable multiplier used to increase or decrease the leadtime-demands component of the minimum requisition objective. It is used in the RXCOMP subroutine in determining "optimal" values of the requisition objective of item for all branches including the central warehouse. In the main program, however, FK1 is used when updating the item requisition objectives for the branches and FK3 is used when updating the item requisition objectives for the central warehouse. This provides a means of using a different multiplier

on the central warehouse than on the other branches if so desired.

769 Sum the product requisition rate at the central warehouse to give a system requisition rate at the central warehouse.

770 Transfer the total of branch requisition objectives (except central warehouse's) to another storage location for later summarization for system total.

771 Compute the number of branch interim reorders per production order cycle.

772-777 Compute the values of little f and k. Capital K is used in determining the optimal requisition objective.

778 Compute the expected value of the cost of a shortage at the central warehouse. Since shortage at the central warehouse may mean a failure to meet direct demand and/or a branch interim order the average value must be taken. This cost factor is further complicated since a failure to meet an interim order may or may not result in an actual shortage at the branch prior to the receipt of the next production allocation.

779 BPIJK is a computational factor representing the probability of ordering.

780-781 RPJK is the value of the optimal requisition objective.

782 Modify the optimal requisition objective value for system and warehouse constraints.

783 Compare the optimal value of requisition objective to total warehouse demands in a shelf-life period and take

minimum quantity.

784 Compare the optimal value of requisition objective to the lower limit and take the maximum. The lower limit is given as the reorder level plus a multiple of the total demands during a production order cycle plus production leadtime period.

785 Add the requisition objective of central warehouse to total for branches to give product requisition objective.

786 Compute the ratio of central warehouse optimal item requisition objective to total direct demands of the item. This expresses the requisition objective in weeks of demands rather than in units. In the main program, the requisition objective is updated by multiplying this factor by the current demand rate of an item to give the current level in units again.

787-788 Round off the product requisition objective to the next higher unit and compute the product reorder level as difference between the product requisition objective and the economic order quantity for the product.

789-799 Sum the product totals to give system totals for:

1. SD - System Direct Demands Rate: All products, all items, all locations.
2. SDB- System Branch Demands Rate: All products, all items, all branches except the central warehouse.
3. SGP- System Goal for Branches: The System

Branch demand rate times the branch goal period or the number or product units required in a specified period (goal).

4. SGPP- System Goal, Plant (Central Warehouse). The system direct demand rate times the plant goal period or the number of product units required in a specified period (goal).

5. SDLB- System Demands in a production Leadtime for Branches only.

6. SDL- System Demands in a production Leadtime all branches and warehouse.

7. SRP- System Requisition Objective for Branches only.

8. SORD- System Order Rate

9. SQ- System Economic Order Quantity

10. SQP- System Economic Order Quantity for Branches only.

11. SR- System Requisition Objective

800 RETURN card - returns the program to the main program to commence simulation run on the product and related items for which the base rules have just been computed.

801-803 Print Summary Section

804-814 Management Data

1. SG - System Goal: sum of branch and plant goals.

2. V-2 Weeks of supply in the branches.

3. V-3 Weeks of supply in system.

4. V-4 Difference between number of weeks of

supply in the branches under the SRI rules and under the alternate rules.

5. V-5 Difference between the number of weeks of supply in the system under the SRI rules and under the alternate rules.

6. W1 - Overall System Branch Goal in weeks.

7. W2 - Overall System Goal in weeks.

8. V8 - System Requisition Objective less System Demands in a production leadtime. This is the approximate maximum system on-hand quantity.

9. V9 - Approximate average on-hand quantity.

10. V10- System Requisition Objective for Branches less system Demands in production lead-time for Branches. This is the approximate maximum on-hand quantity in the branches.

815 PRINT statement for Base Rules Summary

816-819 FORMAT for PRINT statement of card 815.

820-825 FORMAT statements for the PRINT statements used for checkout purposes.

826 END (of SUBROUTINE) card.

SUBROUTINE PLACE BRANCH ORDER (ICON)

827 Definition of Subroutine

828-846 Dimension and place in common those subscripted variables which are used in other subroutines of the program. Note that all subscripted variables used in

this subroutine are in common.

847-848 Test to ensure that the quantity to be ordered is positive. If quantity is zero or negative no action is taken to store the "order" value and program continues.

849-853 Test the branch order quantity against the warehouse on-hand balance to see if order can be filled. If not, program returns to the mainline. However, if the order can be filled the warehouse on-hand balance and stock-position are reduced accordingly and the on-order tally and stock-position of the branch is increased accordingly.

854-855 An order-count tally is kept when using the SRI rules. If it is desired to main this tally for other rules, change the parameter ICON and read in or set the parameter equal to 1.

856-859 The following cost and count tallies are updated for each branch order.

a. The cost of freight. WJ(J) is the weight per unit of the J-th item (lbs. per unit); FRK(K) is the freight cost per hundred weight for the K-th Branch and, of course, the ,01 is a conversion factor from lbs. to cwts.

b. The cost of processing (receiving).

c. Broken pallet count. This count is expressed in the number of units by which the order quantity exceeds the full pallet load. UPCJ(J) is the

number units per case and CPPJ(J) is the number cases per pallets. Their product is the number of units per pallets.

d. Broken case count. This count is also expressed in units; it is the difference between the order quantity and the nearest full case quantity.

e. Cost of picking and loading. This is the cost incident to issuing in non-pallet load or the cost of breaking into a full pallet load and issuing broken lots. CPFP is the unit cost of breaking the full pallet.

860 After updating the necessary cost and count tallies we are ready to place the branch order and its delivery date into the Branch Order Queue. Here we are simulating the delay between the actual placement of an order and the availability of the items at the Branch. The subroutine parameter, N, is set to 1 to route us to the appropriate section of the queuing routine.

861 End of Subroutine card.

SUBROUTINE PREDICT

862 Definition of Subroutine

863-881 Dimension and place in common those subscripted variables which are used in other subroutines of the program. No local subscripted variables are used in this subroutine.

882-883	The predicted weekly demands are basically a nine week moving average. However, since the demand input data is monthly demand, a complicated indexing is required to select the appropriate month in which the most recent nine weeks have occurred and the weighting factor given to each. Each monthly demand rate selected is scaled down to a weekly rate which is then weighted by the number of times the monthly rate is included in the past nine week period. J1 and J2 are variable indexing limits depending on time. The constant values 44 and 52 are chosen so that the first weeks prediction will be based on the 11th and 12th month's demand. (See card 137)
884-887	Initialize total product and total item demand rate and set up do-loops on the items and locations.
888	Test to see if the branch carries the item. If the branch does not carry the item no prediction of demands is necessary. The computer will not go through the prediction routine for this branch thereby saving on computer time.
895-897	If the value of L as computer on card 891 is less than or equal to 12 we test to see if it is zero, if it is we add 12; if the value lies between 1 and 12 inclusively we use that month's demand rate scaled down to a weekly value. Nine such weekly demand rates are accumulated in the variable X.
898-899	The item-location demand rate is averaged over the

previous nine weeks and the item demand rate is the sum of these over all branches.

900 Continue Statement for do-loop on branch. If the branch does not carry the item, the program proceeds to the next branch without waste of time going through the prediction routine for that branch.

901 Sum the demands for each item to give product demand rate. Item demand rates must be converted from units to product-equivalent units.

902 End card for subroutine.

SUBROUTINE RDTP (Read Tape)

903 Definition of Subroutine. All the input data for the product, items, and branches are read into the program by this subroutine. The control parameters are read by a separate read statement at the beginning of the program. The following data is read from the cards:

Product Card

1. The product (Base) identification number
2. Shelf life in months
3. Production leadtime in days
4. Batch size increment in product units
5. Batch size mean in product units
6. Branch Goal (Requisition Level) in days for the Alternate rules.
7. Warehouse Goal (Requisition Level) in days for

the Alternate rules.

Item (Size) Card

1. Size code or item identification number
2. Item conversion factor, i.e., number of product-equivalents per unit of the item (e.g., gallons per quart).
3. Units of the item per case
4. Cases per pallet
5. Unit production cost in cents
6. Unit selling price in cents
7. Weight per unit (pounds per unit) times ten

Branch Card

1. Branch Code Number
2. On-hand balance for each item
3. Inventory position for each item
4. Twelve monthly demand rates

904-906 Dimension and place in common storage all subscripted variables which are used in other subroutines of the program.

907 READ statement for all three types of input cards. The input card all have 17 fields and use a common format statement. The data are first placed into a temporary storage location (AO(K)); after program has returned to the main program, these data are transferred to their permanent location. This device was utilized to standardize the reading of the input data. Originally all the product, item, and

branch data were pre-recorded on magnetic tapes since the volume of data was too large to read into the program efficiently from cards. This procedure could be used again by altering this READ statement to a READ INPUT TAPE statement.

908 Test to see which type of card has just been read. AO(1) is 1 for a product card, 2 for an item card, and 3 for a branch card. If a product card has just been read the AO(1) flag indicates that it is now time to simulate for the previous product.

If an item card has just been read, a positive value in AO(1) will send the computer back to the main program where the subsequent branch cards pertaining to the item will be read.

If a branch card has just been read, a positive value in AO(1) will again send the computer back to the main program where the program will check for additional items under the same product and for branch data under the item.

909 After reading the last branch card of the last item of the last product, the program again calls the RDRT subroutine which attempts to read the next card. Since the last input card is a blank, a zero will be read into the AO(1) location. The test made on this card will cause the value to be reset to a -1. The computer then returns to the main program where the IEOF is set to 2 signifying that the last product is ready

for simulation and to stop the program after the simulation is completed.

910 Return Card

911 Format for READ statement

912 End of subroutine card

SUBROUTINE GENDEM (Generate Demands)

913 Definition of Subroutine

914-932 Dimension and place in common storage all subscripted variables which are used in other subroutines of the program.

933 An index number is computed to determine which monthly demand rate will be used to compute the actual weekly demand rate. The schedule works out as follows.

T (Time Frame)	L (Index for Month)	Frequency of Selection
1,2,3,4,5	1	5
6,7,8,9	2	4
10,11,12,13	3	4
14,15,16,17,18	4	5
19,20,21,22	5	4
23,24,25,26	6	4
27,28,29,30,31	7	5
etc.	etc.	etc.

934-935 Set up do-loops on items and locations.

936 Test to see if branch carries the item.

937 Compute the actual weekly demand rate by scaling the

monthly demand rate down to a weekly rate.

938-939 Continue card for do-loops and End of subroutine card.

SUBROUTINE QUEUE B (N) (Queue Branch Orders)

940 Definition of subroutine

941-945 Dimension Local Subscripted Variables. These local variables are temporary storage locations for the order quantities, delivery dates, item and branch indices. Once computed they must be stored until the proper time to simulate delivery at the branch. One hundred and fifty storage locations have been reserved for these Branch interim orders.

946-963 Dimension and place in common storage all subscripted variables which are used in other subroutines of the program.

964 The subroutine is divided into three sections; the first places the branch order in the queue, the second removes it from the queue, and the third section initializes the queue. This computed GO TO statement directs the program to one of three sections of the subroutine. If the parameter value is 1, the subroutine places the order quantity, delivery date, and indices in the Queue; if the parameter is 2, the subroutine takes the orders out of Queue and simulates receipt of order at the branch; if the parameter value is 3, the subroutine is initialized in preparation for

the next product.

965 The I index counts the number of orders in the queue and also identifies the order.

966 Test to see if the queue is filled. If it is, this information is printed out to advise analyst but program continues which in effect zeros out the order.

967-970 The order, delivery, date, item, and location indices are transferred to Queue.

971 RETURN card

972-973 PRINT statement for Queue overflow condition and RETURN card.

974-979 Test to see if there are any orders in the Queue. If there are no orders (I=0) then the order variable (ORD) is zeroed and program returns to the main program. If there are orders in the queue, a do-loop is set up to screen through all the delivery dates to see if it is time to remove any orders from the queue and to simulate receipt at the branch.

980-982 Transfer the order quantity, item, and location indices to main program locations.

983-986 Transfer the last order to queue to the position just vacated by the order whose delivery date was due. This is done in order to be able to place the next branch order in the last queue position and to conserve running time and to minimize the number of storage locations for the queue.

987 Since the order data of the last queue position has

now been transferred to the vacated slot, the order delivery date (QODD) of the last position is erased.

988 The total number of orders in the queue is reduced to account for the order "delivered".

989 RETURN card

990-991 This is the initialization section of the queue.

992 Format statement for Print statement used to advise of queue overflows.

993 End of Subroutine Card

SUBROUTINE QUEUE P (M) (Queue Production Orders)

994 Definition of Subroutine

995-999 Dimension local subscripted variables.

1000-1017 Dimension and place in common storage those subscripted variables which are used in other subroutines of the program.

1018 The subroutine is divided into three sections; the first places the production order data in the queue, the second removes it from the queue, and the third section initializes the queue. This computed GO TO statement directs the program to the appropriate section as indicated by the parameter value.

1019-1023 Use a do-loop to screen the delivery date locations for an empty slot to store next production order. If all locations are filled, the computer will print "Queue overflow" statement and return to the main

program; the order is, in effect, cancelled.

1024-1029 Transfer the production order data to the queue. Allocations to sizes and the allocations to the warehouse for each item are stored. The machine then returns to the main program.

1030-1034 This is the second section of the subroutine which simulates the delivery of the production orders. Here a do-loop is used to screen the delivery dates in the queue to determine if any are due. If there are no orders in the queue the order is zeroed and the computer returns to the main program.

1035-1038 The production order data is transferred from the queue locations to main program locations.

1039-1040 The delivery date location of the production order removed from the queue is zeroed. Since there are only delivery dates there is no need to maintain the orders in consecutive locations as in the Queue B routine.

1041-1043 Initialize the production order delivery dates locations.

1044-1045 Format for Print Statement which advised of queue overflows and End-of-Subroutine card.

MISCELLANEOUS FUNCTIONS

1046-1048 Definition of RNDNF Function. This function rounds the input quantity to nearest lot size, e.g., rounding an order of so many units to the nearest case lot

quantity.

1049-1051 Definition of RNDCF Function. This function rounds the input quantity to the next higher lot size.

1052-1054 Definition of RNDLF Function. This function rounds the input quantity to the next lower lot size.

1055 End of program card

ALTERNATE DECISION RULES

288-290 Branch Ordering Procedure heading

291-292 Determine whether it is time to conduct a review of the stock records. If it is time, X will be zero. If it is not review time then the program jumps to the next section.

293-300 Set up a do-loop on items to screen the reorder levels against the inventory positions for each item in the warehouse. If the reorder level is less than or equal to the inventory position it is assumed that the warehouse's stock will be sufficient to meet all branch needs and we then test the branch reorder level against their inventory position for each item. If the branch inventory position is below or equal to the reorder level we compute the order quantity as the difference between the branch requisition and branch inventory position. If the warehouse inventory position had been less than its reorder level indicating that it was low on stock of the item then the branch order is reduced by one half.

301 The delivery date is computed by determining the last scheduled delivery date then adding the time period until the next schedule shipment and the in-transit time.

302 The value of the inventory intransit corresponding to this order is added to the previous total.

303 Call the Place Branch Order subroutine to simulate the filling of the branch order at the central warehouse and updating of the stock records. The PBO subroutine calls the Queue B routine to place the Branch order in the queue until the desired delivery date arrives.

304 CONTINUE card

305-307 Production Ordering Procedures heading

308 Initialize the Production Order Batch Size variable.

309-312 Compute the production quantity as the sum of the difference between the requisition level and inventory position at the central warehouse. Production orders are based solely on the item needs at the warehouse under these rules.

313-318 Test to see if production order quantity is positive. If it is the value is transferred to the "ord" storage location and then the production order quantity is rounded to the next higher batch size. To ensure that a minimum economic order quantity is produced, the order quantity is taken as the maximum of the rounded order quantity and a predetermined minimum EOQ.

CHAPTER IV

OUTPUT REPORTS

1. Control and branch card input data

A. Selectors

1. ICON1. Rule Selector: A value of 1 indicates that the SRI decision rules are to be used to compute the requisition levels, reorder levels, and economic order quantities; a value of 2 indicates that the alternate set of decision rules are to be used.
2. ICON2. Not used in the present form of the simulator.
3. ICON3. Stockout Policy Selector: A value of 1 indicates that demands are partially filled, i.e., issues are made until a zero balance is reached; a value of 2 indicates that demands which can not be fully satisfied are cancelled.
4. ICON4. Round to Cartons Selector: A value of 1 indicates that Branch interim orders are not rounded to case lots; a value of 2 indicates that the order quantity is rounded to the next higher full case lot.
5. ICON5. Round to Pallet Selector: A value of 1 indicates that Branch interim orders are not rounded to pallet lots; a value of 2 indicates that the order quantity will be rounded to the next higher full pallet lot if it is economical to do so. To make this decision the holding cost of the additional units is compared to the additional costs incurred due to breaking and handling less than full pallet loads.

B. Costs

1. Opportunity cost (AZR): The cost of inventory investment in

cents per dollar per year.

2. Physical Storage costs (APB and APP): The cost attributable to the physical handling and sheltering of the inventory is given in cents per product-unit per year. A different cost rate may be assigned to the central warehouse than that which is assigned to the branches.

3. Order Preparing cost (CPPO): The administrative cost of preparing and processing a production order in cents per order.

4. Picking and Loading (CPL): The additional cost of the physical handling of broken cases and pallets is given in cents per item-unit handled in a broken case or broken pallet lot. Note that there is no additional cost for handling full pallet lots since this is taken as the standard handling procedure.

5. Cost of Receiving (CR): The costs incurred in the actual receiving, handling-in, and storing of items in cents per hundred weight.

C. Multipliers

1. Demand Forecast: This multiplier may be used to increase or decrease the predicted demands (see card 898) and thereby study the effects of changes in future demand patterns.

2. Lead Time: These multipliers are used to study the effects of increases or decreases on the branch interim reorder lead-times and on the production-run component of the production order leadtime. (See card 76)

3. Goals: Multipliers used to vary the branch and central warehouse requisitioning goals. (See cards 74 and 75)

4. Standard Cost: Multiplier used to vary the standard unit production cost. (See card 102)

D. Leadtime Components

1. Reaction (LR): The time between the reaching of a reorder level and the recognition of that fact and the initiation of action to place an order. The input is in days.

2. Communcation (LC): The time between reaching a decision to place a production order and the receipt of that order by the production unit. The input is in days.

3. In Transit (LS): The shipping time between the central warehouse and a branch. In the simulator it is assumed for simplicity that the shipping time is the same to all branches. No provision is made for emergency shipments. The input is in days.

E. Base Rules Computation Parameters

1. AZR, APB, APP, CR: See section B above.

2. LR, LC, LS: See section D above.

3. A1: The cost of setting up a production run in dollars per set up.

4. A3: Not used in the present form of the simulator.

5. MU: The system inventory constraint factor. This multiplier is used to reduce the "optimal" requisition objective of each item-location by some multiple of the demands in a branch production leadtime. (See cards 707 and 782). It is also applied to reduce the economic order quantity (see card 665). By varying the value of MU the system average holding cost or system average on hand quantity can be set to any pre-determined

value. Setting MU equal to zero is equivalent to placing no constraints on the system.

6. K6: Multiplier used to increase or decrease the central warehouse item reorder level in the RXCOMP subroutine. (See card 755).

7. K5: Multiplier used to increase or decrease the system item-reorder level in the RXCOMP subroutine. (See card 751).

8. K4: Multiplier used to increase or decrease the item-location minimum requisition objective. The minimum level is taken as the reorder level plus the demands during the production cycle plus production leadtime period. The multiplier increases only the later component of the minimum requisition objective. Hence, doubling the multiplier will not necessarily double the minimum requisition level. This multiplier applies to the central warehouse's minimum level as well as to the other branches. (See cards 720 and 784 in RXCOMP subroutine). It should be noted that this multiplier is used as a dummy variable within the program and is not an input quantity. Prior to computing the "optimal" requisition objective level for the central warehouse, the value of the multiplier FK3 is put into FK4 (card 768), and prior to computing the level for the other branches the value of FK1 is transferred to FK4 (card 680). It is not known why the variables FK1 and FK3 were not used directly in computing the minimum requisition objectives.

9. K3: This multiplier is used in the main program (card 145) to update the "optimal" central warehouse requisition level. It is used to increase or decrease the lower limit on the

requisition level. See K4 for further information.

10. K2: Multiplier used to increase or decrease the item-location reorder level for all locations except the central warehouse (see card 640, 719, and 151). By varying this multiplier the lower limit on the requisition objective is also changed since the minimum requisition level depends on the reorder level.

11. K1: This multiplier is used in the main program (card 153) to update the "optimal" item-location (except for central warehouse) requisition objective. It is used to increase or decrease the lower limit on the requisition level. See K4 for further information.

12. CM2: Cost multiplier used to increase or decrease the production setup cost (card 665).

13. VLP1: The rel-variance (ratio of variance to the square of the mean of the production-run component of leadtime). It is a dimensionless quantity.

14. VLP3: Not used in the present form of the simulator.

F. Branch Data

1. Code: The activity accounting numbers assigned to the branches. Code 1 is assigned to the central warehouse.

2. Cost of Shipment: The freight rate in cents per hundred weight. The freight rate at the central warehouse is given the negative value of the receiving cost in order that the unit cost at the warehouse will be the unit production cost.

3. Scheduled Shipping Frequency: The period between shipments to each branch.

4. Branch Inventory Constraints Multipliers: If it is desired

to limit the individual branch total average inventory to some specified dollar value or on-hand quantity, vary this parameter on a trial and error basis until the proper setting is obtained. Each branch can be given a different constraint factor if it is so desired.

2. The item, product, and system summary report

The average inventory values, sales, costs, and stock effectiveness data are summarized here by item, product, and system. Complete branch data is listed for each item. The "Holding Cost" given in the report includes the opportunity cost factor as well as the physical holding cost. The "Order Count" is the number of interim branch orders for the item. The "Estd Cost" of stockouts is the profit loss due to lost sales. The number of broken cartons and pallets is given in item units rather than in the number of cartons or pallets actually opened to fill an order. The other headings on the report are self-explanatory.

3. The branch summary report

The inventory values, cost data, and stock effectiveness measures are summarized by branch for all items and products.

All the data given in this report under the "Inventory Values" heading including the stock turn ratio are expressed in terms of dollars. The "Order Count" is the number of scheduled deliveries to the branches. The "Preparation and Processing" cost is the cost of making the scheduled deliveries to the branches. It is the number of scheduled deliveries per year times the cost of preparing and processing an order (CPPO). The "Picking and Loading" cost is the sum of the costs incident to handling

of broken cartons and broken pallet lots and the cost of handling the full pallet lot portion of the orders. The "Unloading Processing" cost is the cost of receiving and storing materials at the branch. The other headings on the report are self-explanatory.

4. The base rules summary report

1. D: The expected demands per week for the system.
2. RPR/D: The requisition objective in weeks for all branches except the central warehouse under the SRI rules.
3. R/D: The requisition objective in weeks for all branches including the central warehouse under the SRI rules.
4. APR/D and A/D: Same as 2 and 3 above except these are for the alternate policy rules.
5. (RPR-APR)/D and (R-A)/D: The difference between the requisition objective requirements under the SRI and the alternate policy rules.
6. DL(PR): The "pipeline" or leadtime quantity in product units for all branches except the central warehouse.
7. (DL): The "pipeline" or leadtime quantity in product units including the central warehouse.
8. (ON-HND-MX): The average maximum on-hand level for all branches including the central warehouse.
9. (ON-HND-MX)PR: Same as 8 above but does not include the central warehouse.
10. (ON-HND-AVE): The expected on-hand inventory level for all branches including the central warehouse.
11. (ON-HND-AVE)PR: Same as 9 except the central warehouse is not included.

12. ORD: The average number of production orders per week.

13. Branch Picking Cost: The system-wide cost of picking and loading for both branch interim orders and production allocations.

14. Production Cost: The total set-up costs.

15. Shortage Cost: The loss of profit due to nonavailability of items to meet demands as they occurred.

16. Holding Cost: The physical holding costs incurred at all branches including the central warehouse.

17. Average Value in Transit: The dollar value of the average number of units in transit between the central warehouse and the branches.

CONTROL CARD INPUTS -- RUN NUMBER 5

SELECTORS --	RULE	1
	STOCKOUT	1
	ROUND TO CASES	2
	ROUND TO PALLETS	2
COSTS --	OPPORTUNITY	.205
	PHYSICAL STORAGE - BRANCHES	.24
	- WAREHOUSE	.43
	ORDER PREPARING	.75
	PICKING AND LOADING - BROKEN PALLETS	.07
	- BROKEN CASES	.23
	- FULL PALLETS	.12
	RECEIVING	.15
MULTIPLIERS --	DEMAND FORECAST	1.00
	LEADTIME - WHS TO BRANCH	1.00
	- PLANT TO WHS	1.00
	GOALS - BRANCH	1.00
	- WAREHOUSE	1.00
	STANDARD COST	1.00
LEADTIME --	REACTION	4.
	COMMUNICATION	1.
	IN TRANSIT	11.

BASE RULES COMPUTATION

AZR	.205	APB	.240	APP	.430	CR	.150
LR	4.000	LC	1.000	LS	11.000	A1	400.000
A3	.000	MU	.000	K6	1.000	K5	1.000
K1	1.000	K2	1.000	K3	1.000	CM2	1.000
VLP1	.000	VLP3	1.000				

BRANCH DATA

CODE 1. 45. 57. 62. 63. 64. 68. 88. 106. 107. 111. 122. 127. 136. 138. 150. 175. 180.

CS -.15 .83 .42 .53 .39 .67 .37 .87 .56 .34 .26 .23 .32 .43 .49 .32 .35 .47

W 0. 2. 1. 2. 1. 1. 1. 1. 1. 2. 2. 2. 1. 1. 3. 2. 3. 2. 1. 2. 3. 1. 1. 1.

MUK .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0 .0

CHAPTER V

CONTROL CARD FORMATS

Card 1. This card contains the run number, the control selectors settings and the system cost factors.

a. Columns 1-5. The letters "IDENT" are placed in these columns to identify the run number. If desired these columns and/or any columns on subsequent cards used for alphabetical identification may be left blank.

b. Columns 6-8. These are usually left blank but may be utilized for any non-input data as the letters "IDENT" in columns 1-5.

c. Columns 9-14. The run number designator is entered here. It may be an alphanumeric designator.

d. Columns 15-24. The letters "SELECTORS" are placed in the columns, right adjusted to leave column 15 blank.

e. Columns 25-26, 27-28, 29-30, 31-32, 33-34. The values for ICON1, ICON2, ICON3, ICON4, and ICON5 (i.e., the number 1 or 2) are placed in these columns, right adjusted.

f. Columns 35-40. The letters "COSTS" are inserted here, also right adjusted.

g. Columns 41-44. The opportunity cost (AZR) is inputed in mils, right adjusted. A value of 205, for example, indicates an opportunity cost of 20.5 cents per dollar of investment per year.

h. Columns 45-58. The physical holding cost for the branches (APB) is inputed in cents, right adjusted. A value of 24, for example, indicates a holding cost of 24 cents per product unit

per year.

i. Columns 49-52. The physical holding cost for the central warehouse (APP) is inputed in cents, right adjusted. A value of 43, for example, indicates a holding cost at the warehouse of 43 cents per product unit per year.

j. Columns 53-57. The cost of preparing and processing an order (CPPO) is placed here in cents, right adjusted. The value of 75, for example, indicates a cost of 75 cents per order.

k. Columns 58-62. The cost of handling broken pallets (CBPC) is given here in cents, right adjusted. The value of 7, for example, indicates a cost of 7 cents per item unit handled.

l. Columns 63-67. Same as k above except for broken cases (CBCC).

m. Columns 68-72. The cost of receiving (CR) in cents per hundred weight, right adjusted.

n. Columns 73-77. The cost of handling (or picking) a full pallet lot (CPFP) given in cents per pallet, right adjusted.

Card 2. This card contains the management multipliers, the reaction, communication, and shipping components of the production leadtime and the cost of a production set-up.

a. Columns 1-11. The letters "MULTIPLIERS" is inserted here for identification of the card and the subsequent input data.

b. Columns 12-15, 16-19, 20-23, 24-27, 28-31, 32-35. The values for the demand forecast (FMDF), the warehouse to branch (FMLWB), and the plant to warehouse leadtime (FMLP) multipliers, the branch and plant goal multipliers (FMPG, FMBG), and the standard cost multiplier FMCS are placed in these columns in the given order. The multipliers are specified to two decimal

places within the program; hence, a value of 100 indicates a multiplier of 1.00. A setting of 1.00 for the multipliers is equivalent to not using them. They should never be set at zero.

c. Columns 36-45. Leave blank

d. Columns 46-55. The words "LEAD TIMES" are inserted here.

e. Columns 56-58, 59-61. The value of the reaction (FLR) and communication (FLC) components of leadtime are given here in days, right adjusted.

f. Columns 62-64. Leave blank

g. Columns 65-67. The value of the shipping or in-transit component of leadtime (FLS) is given here in days, right adjusted.

h. Columns 68-72. The letters "A1" indicating the set-up cost are placed here, right adjusted.

i. Columns 73-75. The value of the set-up cost is inputed here in dollars per set up, right adjusted.

Card 3. This card contains the input data for the system capacity constraint (FMU), the reorder level multipliers (FK6, FK5), the minimum requisition objective multipliers (FK1, FK2, FK3), which are referred to as "floors" in [1.], the set up cost multiplier (CM2), and the rel-variance for the production-run component of leadtime (VLP).

a. Columns 1-3. The letters "MU", left adjusted, are listed here for identification only.

b. Columns 4-8. The value of the system capacity constraint (FMU) is given here to three decimal places (i.e., a value of 8300

is read in as 8.300).

c. Columns 9-24. The letters "TRIGGER LEVELS" are entered here, right adjusted, for identification only.

d. Columns 25-27, 28-30. The values of FK6 and FK5 are entered here to one decimal place, (i.e., a value of 14 is read as 1.4). They should never be set to zero.

e. Columns 31-38. The letters "FLOORS" are placed here, right adjusted, for identification only.

f. Columns 39-41, 42-44, 45-47. The values of FK1, FK2, FK3 are entered in these fields in their respective order. The multipliers are given to one decimal place (i.e., a value of 060 is read as 6.0). If these multipliers are set at 1.0 it is equivalent to not utilizing them. The multiplier FK2 should never be set to zero.

g. Columns 48-59. The letters "VARIATIONS", right adjusted, are placed here for identification only.

h. Columns 60-63. The value of the set-up cost multipliers (CM2) is entered here to two decimal places, (i.e., a value of 404 is read as 4.04).

i. Columns 64-71. The value of the rel-variance of the production-run component of the production leadtime is entered here to two decimal places, right adjusted. A value of 150 is read as 1.50.

Card 4. Branch Data heading card to identify subsequent input cards. (Note: The READ format statement is designed to skip a card at this point and would mis-read all the subsequent cards if this particular one is omitted)

Card 5. Branch Code Card. Three columns are used for each branch code with the code numbers right adjusted. Any set of code numbers (each not exceeding three digits) may be used. There is a limitation of 25 branch code numbers.

Card 6. Freight Rate Card. The cost of shipping, FRK(K), in cents per hundred weight is entered for each branch in same order as the branch codes are entered. Three columns are allowed for each freight rate cost. The limitation is also 25 entries, with the numbers right adjusted.

Card 7. Shipping Schedule Card. The scheduled shipping frequency for each branch, FSK(K), is listed on this card, (i.e., the number of weeks between shipments from the central warehouse to the individual branches. Three columns are allowed for each field. The values are given as integers and are right adjusted.

Card 8. The Branch Capacity Constraints. The individual branch constraint multipliers (FMUK(K)) are entered here to one decimal place. Three columns are allowed for each entry and numbers are right adjusted. A value setting of zero is equivalent to placing no constraints on the branches.

Card 9. Product Data Card

a. Column 1. The number 1 is always placed here to indicate that this is a product data card.

b. Columns 2-5. The base identification number of the product is entered here.

c. Columns 6-20. Leave blank

d. Columns 21-25. Enter the shelf-life in months, right adjusted.

e. Columns 26-50. Leave blank

f. Columns 51-55. Enter the length of a production run (FLP) in days, right adjusted.

g. Columns 56-60. Enter the batch size increment (BSI) in product units, right adjusted.

h. Columns 61-65. Leave blank.

i. Columns 66-70. Enter the minimum economic batch size (BSM) in product units, right adjusted.

j. Columns 71-75. The Branch Goal (BG) is given in days, right adjusted.

k. Columns 75-80. The Plant Goal (PG) value is given in days, right adjusted.

Card 10. Item Data Card

a. Column 1. The number 2 is always placed here to indicate that this is an item data card.

b. Columns 6-9. The product base identification number is entered here, right adjusted.

c. Columns 10-11. The item identification number is entered here, right adjusted.

d. Columns 15-20. The item-unit to product-units conversion factor (or size factor) is entered here to four decimal places. The value of 11057 is read as 1.1057.

e. Columns 51-55. The number of units per case (UPCJ(J)) of the item is entered here, right adjusted.

f. Columns 56-60. The number of cases per pallet for the item (CPPJ(J)) is entered here, right adjusted.

g. Columns 61-65. The unit production cost (CJ(J)) of the item in cents is entered here, right adjusted.

h. Columns 66-70. The sales price of the item per unit (PJ(J)) is entered here, right adjusted.

i. Columns 71-75. The weight per unit (WJ(J)) of the item in pounds is entered here, right adjusted.

j. Note: All other columns are left blank

Card 11. The Item-Location Data Card

a. Column 1. The number 3 is always entered here to indicate that this is an item-location data card.

b. Columns 12-14. The branch code (BCK(K)) is entered here, right adjusted.

c. Columns 15-20. The initial on hand balance in item units (OHJK(J,K)) is entered here, right adjusted.

d. Columns 21-25, ---, 76-80. These are 12 fields of five columns each in which the monthly demands are entered in item units. The average long run demands based on these numbers will be used to make the initial forecast of demands and to compute the initial optimal inventory policies. They are also used individually to compute the actual weekly demands lodged at the branches.

e. Note: All other columns are left blank.

CONTROL CARD SAMPLES

```
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
IDENT     SAMPLE SELECTORS 1 1 1 2 2 COSTS 205  24  43  75     7  23  15  12
MULTIPLIERS 100 100 100 100 100 100          LEAD TIMES  4 1     11  A1400
MU  0000  TRIGGER LEVELS 10 10  FLOORS 10 10 10 VARIATIONS 100      100
BRANCH DATA
  1 45 57 62 63 64 68 38106107111122127136138142176189226228
-15 81 41 53 39 67 37 37 56 34 26 23 32 25 32 43 49 32 46 50
  0  2  1  2  1  1  1  1  2  2  1  2  2  2  1  2  2  1  2  2
-60 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
11207                  24                            21  100       500   60   60
2    120701    10000                                  8   20  100  300    5
3             1  300  100  100  100  100  100  100  100  100  100  100  100  100
3            45  500  200  200  200  200  200  200  200  200  200  200  200  200
3           127  600  300  300  300  300  300  300  300  300  300  300  300  300
```

CHAPTER VI

PROBABILISTIC DEMAND PATTERNS

Four subroutines have been designed to utilize probabilistic demand patterns for the SRI simulator. These subroutines will provide 12 random numbers from the appropriate distribution desired for each of the monthly demands. The procedures for scaling the monthly demands to arrive at the predicted and actual weekly demands remain the same.

Normal, exponential, uniform, and Poisson distributions are approximated in these subroutines. The distribution parameters are defined within the subroutine and can be changed to any desired value by changing the appropriate cards. The distributions are truncated to prevent unrealistic values of demand from occurring. In the case of the continuous functions, the random numbers are rounded to the nearest integer. Even with these modifications a close approximation to the real distributions has been attained which is more than accurate for the purposes intended. The normal distribution is truncated two standard deviations above and below the mean. The exponential distribution is truncated at that value above which only a certain desired probability (error term) occurs. The uniform distribution, of course, is bounded by its upper and lower end points which may be set to any desired values. The Poisson distribution is terminated at three times its mean. Table 1 gives the results of some test runs on these subroutines.

In order to use the random demand subroutines the program and input cards must be modified as follows:

1. Input Cards. On each branch data card leave columns 21 through 80 blank instead of inserting the monthly demand values.

2. Subroutine Parameter Cards. Make new cards with the desired values of the distribution parameters and in the case of the exponential subroutine set the desired value for "ERROR", i.e., the probability area to be truncated (a value of .01 to .05 is suggested).

3. COMMON Cards. All COMMON cards of the main program must be listed in the subroutine and similarly all COMMON cards of the subroutine must be listed in the main program.

4. Call Cards

   a. Initialize the Random Number Generator. In order to vary the random numbers used in different runs the random demand subroutine may be called a few times at the beginning of each run by the following routine:

```
ISPIN = ( ANY DESIRED FIX POINT NUMBER)     65A
CALL UNIFORM (1)                            65B
DO 99 K = 2, ISPIN                          65C
99 CALL UNIFORM (2)                         65D
```

   b. Call for the Monthly Demands. Insert the distribution call card after card number 118; for example:

```
CALL NORMAL (2)                            118A
```

The above procedures will introduce only one random demand pattern for all products and items since the distribution parameters are fixed for each simulation run. If it is desired to vary the parameters for different products, items, or locations, this may be accomplished by listing the parameters on either the product, item or branch data card. Since each distribution has two parameters columns 26-30 and 31-35 of the data cards may be used for this purpose. The RDTP subroutine will

place these values in the AO(15) and AO(16) locations.

Products: To vary the parameter values for each product, use the following routine:

1. Insert the value of the two parameters in columns 26-30 and 31-35 in the product data card, right adjusted. These values will be read as integers and must be scaled if decimal places are desired.
2. Place the names of the parameters in COMMON and remove the parameter cards from the subroutine.
3. Prepare cards for the main program to transfer the parameter values from the AO array to their permanent storage locations. For example, if using the SUBROUTINE NORMAL, prepare the following cards:

```
FMEAN = AO(15)                  73A
STDDEV = AO(16)                 73B
STDDEV = STDDEV * .01           73C
```

Insert these cards after card 73 in the main program. The last card is an example of scaling a value to a desired decimal figure and may or may not be required.

Items: To vary the parameter values for different items under the same product, use the following procedure:

1. Same as (1) for products except insert parameters values on the item data cards instead of the product data cards.
2. Same as (2) for products
3. Same as (3) for products except number the transfer cards 98A, 98B, and 98C and insert these cards after card 98 in the main program.

Items-locations: To vary the parameter values for each location which carries the item, use the following procedure:

1. Same as (1) for product except insert the parameter values on the branch data card instead of the product data card.
2. Same as (2) for products.
3. Same as (3) for products except number the transfer cards 116A, 116B, 116C and insert these cards after card 116 in the main program.

SUBROUTINE UNIFORM. Uniformally distributed random numbers are generated on the interval (A,B) by the formula,

$$U(A,B) = A + (B-A)*RN$$

where RN is a U(0,1) random number produced by SUBROUTINE RAND. The above formula was derived by equating the integral of the uniform density between the lower end point and a variable upper limit to the U(0,1) random number generated by RAND and solving for the variable upper limit. The variable "FLWPT" is the lower end point in the floating point mode and the variable "UPENDPT" is the upper end point, also in the floating point mode. These variables may be set at any desired values.

SUBROUTINE EXPONTL. Exponentially distributed random numbers are computed using the formula,

$$Y = \frac{1}{\theta} \mathrm{Log}_e \left(1./(1. - RN) \right)$$

where RN is a U(0,1) random number produced by SUBROUTINE RAND. This formula was also derived by the integral method. The variable "THETA" is the parameter of the distribution and not the mean which is, of course, the reciprocal of the parameter. The variable "ERROR", used

elsewhere in the subroutine, is the value of the probability area under the density curve truncated in order to rule out unrealistic demands.

SUBROUTINE NORMAL. The normally distributed numbers are computed by the well known method of summing 12 U(0,1) random numbers, subtracting the number six, and multiplying this quantity by the desired standard deviation and then adding the desired mean. The variable "FMEAN" and "STDDEV" are the mean and standard deviation desired for the distribution. Since it is possible to generate negative demands with this distribution the standard deviation and mean chosen should be such that the mean less twice the standard deviation is greater than zero. On the other hand, negative demands could be allowed to simulate materials returned to store or items to be exchanged.

SUBROUTINE POISSON. The discrete analog of the integral method is used for this distribution. The probability mass function is summed term by term and compared to the U(0,1) random number obtained from the RAND subroutine. The Poisson number is taken to be the integer corresponding to that term which when added to the sum of the previous terms exceeds the U(0,1) random number. The variable "FLAMBDA" is the mean of the distribution.

```
      SUBROUTINE NORMAL (ICON6)
C
C
      COMMON  INDX, NX, NRNC, NXIN, NRC, NSTRN
C     INSERT ALL THE DIMENSION AND COMMON CARDS OF THE MAIN PROGRAM
      GO TO (1,2), ICON6
    1 ITRAN = 1
      NRNC = 10000
      NRC = 0
      NSTRN = 0
      INDX = 6
    2 FMEAN = 50.0
      STDDEV = 25.0
      UPLIMIT = FMEAN + 2.*STDDEV
      FLWPT = FMEAN -2.*STDDEV
      DO 38 J = 1,12
      M = 13 +J
      Y = 0.0
      DO 5 L=1,12
      CALL RAND (ITRAN)
      RN = FLOATF(NXIN)/10000.
    5 Y = Y + RN
      AO(M) = FMEAN + STDDEV*(Y-6.)
      IF(AO(M) -FLWPT) 30,30,35
   30 AO(M) = FLWPT
      GO TO 38
   35 IF(AO(M) - UPLIMIT)38,36,36
```

```
36 AO(M) = UPLIMIT
38 AO(M) = INTF( AO(M) +.5)
   RETURN
   END
C
```

```
C
C
      SUBROUTINE EXPONTL (ICON6)
C
C     INSERT ALL THE DIMENSION AND COMMON CARDS OF THE MAIN PROGRAM
      COMMON  INDX, NX, NRNC, NXIN, NRC, NSTRN
      GO TO (1,2), ICON6
    1 ITRAN = 1
      NRC   = 0
      NSTRN = 0
      INDX = 6
    2 NRNC = 10000
      THETA = .01
      ERROR = 0.01
      UPLIMIT = LOGF ( 1./ ERROR ) / THETA
      UPLIMIT = INTF( UPLIMIT )
      DO 5 J=1,12
      CALL RAND (ITRAN)
      RN    = FLOATF (NXIN) / 10000.
      M = 13 + J
      AO(M) = INTF( LOGF(1./(1. - RN))/THETA  +0.5)
    5 AO(M) = MIN1F ( AO(M),  UPLIMIT )
      RETURN
      END
```

```
C
C
      SUBROUTINE POISSON (ICON 6)
C
C     INSERT ALL THE DIMENSION AND COMMON CARDS OF THE MAIN PROGRAM
      COMMON INDX,NX,NRNC,NXIN,NRC,NSTRN
      GO TO (1,2), ICON6
    1 ITRAN = 1
      INDX = 6
      NSTRN = 0
      NRC = 0
    2 NRNC = 10000
      FLAMBDA = 4.
      IUPLIM = 12
      DO 11 J = 1,12
      CALL RAND (ITRAN)
      RN    = FLOATF (NXIN) / 10000.
      TESTNR =RN   *EXPF(FLAMBDA)
      M=13+J
      IF(TESTNR-1.)5,5,8
    5 AO(M)=0.0
      GO TO 11
    8 PRODUCT=1.0
      Y=1.0
      DO 9 K = 1, IUPLIM
      XNR1 = FLOATF (K)
      PRODUCT = PRODUCT *XNR1
```

```
      Y = FLAMBDA**K/PRODUCT + Y
      IF(TESTNR-Y)10,10,9
    9 CONTINUE
   10 AO(M) =FLOATF( K )
   11 CONTINUE
      RETURN
      END
C
```

```
C
C
      SUBROUTINE UNIFORM (ICON6)
C
C     COMMON STORAGE
C     INSERT ALL THE DIMENSION AND COMMON CARDS OF THE MAIN PROGRAM
      COMMON INDX, NX, NRNC, NXIN, NRC, NSTRN
      GO TO (1,2), ICON6
    1 ITRAN = 1
      NRC = 0
      NSTRN = 0
      INDX = 6
    2 NRNC = 10000
      FLWPT = 50.0
      UPENDPT = 80.0
      DO 5  J = 1,12
      CALL RAND (ITRAN)
      RN = FLOATF (NXIN) / 10000.
      M = 13 + J
    5 AO(M) = INTF( FLWPT +(UPENDPT - FLWPT)*RN + 0.5)
      RETURN
      END
C
```

```
C
C
      SUBROUTINE RAND (ITRAN)
C
C     RANDOM NUMBER GENERATOR
C
C     LOCAL STORAGE
      DIMENSION XI(8),NXI(8)
C     INSERT ALL THE DIMENSION AND COMMON CARDS OF THE MAIN PROGRAM
      COMMON INDX,NX,NRNC,NXIN,NRC,NSTRN
      EQUIVALENCE (XI,NXI),(I1,NT1),(INDX,DX),(D1,ND1)
C
      GO TO (10,20),ITRAN
   10 ITRAN=2
B     XI(1)=3514 6016 2524 6131
B     XI(2)=0337 1363 2712 7740
B     XI(3)=1760 3011 0710 3016
B     XI(4)=0670 1154 5656 1316
B     XI(5)=1506 7663 7414 1566
B     XI(6)=2100 7160 3140 2631
B     XI(7)=1037 4327 7340 4007
C     XI(8)=0
C
   20 NRC=NRC+1
      L=INDX+1
      ND1=L
B     DX=D1*7
```

```
      M=INDX+1
       ND1=M
B     T1=D1*7
      N=NT1+1
      NXI(M)=NXI(L)+NXI(N)
B     XI(M)=XI(M)*3777 7777 7777 7777
      NX=NXI(M)
   30 LDA(NX),MUF(NRNC),STA(NXIN).
C
      RETURN
      END
C
```

TABLE 1

RESULTS FROM TEST RUNS ON THE RANDOM DEMAND SUBROUTINES

NORMAL DISTRIBUTION — SAMPLE SIZE 500

MEAN	XBAR	STDDEV	SAMPLE VAR	$\sqrt{S.V.}$
25	26.54	25	550.1	23.4
50	49.44	25	615.6	24.8
75	74.79	25	561.3	23.7
100	101.71	25	547.7	23.4
125	124.34	25	506.99	22.5
150	149.50	25	511.69	22.6

EXPONENTIAL DISTRIBUTION — SAMPLE SIZE 500

MEAN	XBAR	STDDEV	SAMPLE VAR	$\sqrt{S.V.}$
100	102.6	100	9,657.9	98.30
20	19.78	20	357.82	18.90
10	9.89	10	88.86	9.42
4	4.04	4	15.16	3.90
2	1.88	2	3.46	1.83

POISSON DISTRIBUTION — SAMPLE SIZE 500

MEAN	XBAR	STDDEV	SAMPLE VAR	$\sqrt{S.V.}$
4	4.10	2.00	3.99	2.00
5	5.06	2.24	5.27	2.29
6	5.92	2.45	5.10	2.26
7	7.10	2.64	7.15	2.67
8	7.83	2.83	7.51	2.74
9	9.06	3.00	9.54	3.09
10	9.98	3.16	9.74	3.12

UNIFORM DISTRIBUTION SAMPLE SIZE 500

INTERVAL	MEAN	XBAR	STDDEV	SAMPLE VAR	$\sqrt{S.V.}$
(0,100)	50	50.91	28.86	823.59	28.69
(10,90)	50	49.81	23.08	526.92	22.95
(20,80)	50	49.07	17.32	273.34	16.53
(30,70)	50	49.74	11.53	135.28	11.62
(40,60)	50	49.13	5.77	32.13	5.66

BIBLIOGRAPHY

1. Stanford Research Institute. Integrated Production-Distribution Control Rules for a Two-Level Distribution System, by Richard H. Davis. May 1963. SRI Technical Memorandum No. 3.

2. Stanford Research Institute. A Multi-Stockpile, Single-Level Distribution System, by Richard H. Davis. SRI Technical Memorandum No. 2.

3. Stanford Research Institute. Modifications to the Interim Model Procurement Rules, by Richard H. Davis. January 1961. SRI Technical Memorandum No. 1; NONR-2311(00).

4. Stanford Research Institute. Item Balance Factor in the Interim Model Procurement Rules, by Richard H. Davis. January 1961. SRI Technical Memorandum No. 2; NONR-2311(00).

5. Stanford Research Institute. The Development and Implementation of Advanced Inventory Control Procedures, by Peter H. Butterfield, etc. al. December 1960.

6. Stanford Research Institute. A Survey of Research Work on the Problems of Physical Distribution. December 1962. SRI General Research Project 004531-812.

7. Stanford Research Institute. Operating Manuel for the Inventory Rules Evaluator for the Aviation Supply Office, by Ben Lefkowitz and Donato A. D'Esopo. November 1961. SRI Project No. IMU-2109.

8. Stanford Research Institute. The Inventory Rules Evaluator, by Ben Lefkowitz and Donato H. D'Esopo. December 1960. SRI Project No. IU-2109; NONR-2311(00).

APPENDIX I

```
      PROGRAM  INVSIM
C
C     LOCAL STORAGE
C
      DIMENSION BCK(25), CHK(25), CLK(25), CPIJK(20,25), DSIJK(20,25),
     1          FITEM(20), RIJ(20), RIJK(20,25), SC(20), SIJ(20),
     2          SOCJK(20,25), SOWJK(20,25), SOWK(25), SYJK(20,25),
     3          VDSK(25), VSYK(25), VZDK(25), XIJK(20,25), ZDIJK(20,25)
C
C     COMMON STORAGE
C
      DIMENSION AO(26), BCCJK(20,25), BF(20,25), BPCJK(20,25),
     1          CFK(25), CJ(20), CPK(25), CPLK(25), CPPJ(20), DIJ(20),
     2          DIJK(20,25), DM(20,25,12), DWIJK(20,25), FLIK(25),
     3          FLSK(25), FMUK(25), FSK(25), FRK(25), OCJK(20,25),
     4          OHJK(20,25), OOJK(20,25), PJ(20), POAS(20),
     5          POAW(20), SIJK(20,25), SJ(20), UPCJ(20), WJ(20),
     6          XDIJ(20), XDIJK(20,25), XK2IK(20), XRIJK(20,25),
     7          XRIJZ(20), XXIJ(20), XXIJZ(20), XIJ(20)
      COMMON    AO, A, APB, APP, AZR, BCCJK, BF, BG, BPCJK, CFK, CJ,
     1          CM2, CPK, CPLK, CPPJ, CR, DI, DIJ, DIJK, DM, DWIJK,
     2          FLIK, FLSK, FMU, FMUK, FLP, FLRI, FK1, FK2, FK3, FLC,
     3          FK4, FK5, FK6, FSK, FRK, J, K, NJ, NK, OCJK, ODD,
     4          OHJK, OOJK, ORD, PG, PJ, POBS, PODD, POAS, POAW, QI,
     5          SIJK, SJ, T, TX, UPCJ, WJ, XDIJ, XDIJK, XK2IK, XIJ,
     6          XRIJK, XRIJZ, XXIJ, XXIJZ
C
C     START OF PROGRAM
C
      RGX=0.25
      NK = 25
      FLRI = 1.
      PAGE = 1.
      Z = 0.
      FNOPR = 0.
      VYSIT = 0.
      IEOF = 1
      READ 1,RUNNO,ICON1,ICON2,ICON3,ICON4,ICON5,AZR,APB,APP,CPPO,CBPC,
     1     CBCC,CR,CPFP,FMDF,FMLWB,FMLP,FMBG,FMPG,FMCJ,FLR,FLC,FLS,A,
     2     FMU,FK6,FK5,FK1,FK2,FK3,CM2,VLP,(BCK(K),K=1,NK),
     3     (FRK(K),K=1,NK),(FSK(K),K=1,NK),(FMUK(K),K=1,NK)
      AZR = AZR * .1
      PRINT 3,RUNNO,ICON1,ICON3,ICON4,ICON5,AZR,APB,APP,CPPO,CBPC,CBCC,
     1     CPFP,CR,FMDF,FMLWB,FMLP,FMBG,FMPG,FMCJ,FLR,FLC,FLS
      PRINT 2,AZR,APB,APP,CR,FLR,FLC,FLS,A,Z,FMU,FK6,FK5,FK1,FK2,
     1     FK3,CM2,Z,VLP,(BCK(K),K=1,NK),(FRK(K),K=1,NK),
     2     (FSK(K),K=1,NK),(FMUK(K),K=1,NK)
```

```
      FLC = FLC / 7.
      FLS = FLS / 7.
      FLR = FLR / 7.
      CALL RXCOMP (1)
      CALL RDTP
      DO 100 K = 1,NK
  100 FLSK(K) = FLS
      FLSK(1) = 0.
      DO 101 K = 1,NK
      VDSK(K) = 0.
      VSYK(K) = 0.
      CHK(K)  = 0.
      CPLK(K) = 0.
      CFK(K)  = 0.
      CPK(K)  = 0.
      VZDK(K) = 0.
      SOWK(K) = 0.
  101 CLK(K)  = 0.
C
C     START NEW PRODUCT
C
  200 T = 0.
      SI = 0.
      VSIT = 0.
      FLPDTM = 0.
      TX = AO(14) * 4.3333
      BG = AO(24) / 7. * FMBG
      PG = AO(25) / 7. * FMPG
      FLP = AO(20) / 7. * FMLP
      BSI = AO(21)
      BSM = AO(23)
      BASE = AO(2)
      CALL RDTP
      NJ = 0
  201 NJ = NJ + 1
      SIJ(NJ) = 0.
      DO 202 K = 1,NK
      DSIJK(NJ,K) = 0.
      OCJK(NJ,K)  = 0.
      SOWJK(NJ,K) = 0.
      SOCJK(NJ,K) = 0.
      BCCJK(NJ,K) = 0.
      BPCJK(NJ,K) = 0.
      OOJK(NJ,K)  = 0.
      CPIJK(NJ,K) = 0.
      SYJK(NJ,K)  = 0.
      ZDIJK(NJ,K) = 0.
  202 BF(NJ,K)    = 0.
```

```
      X = INTF(AO(3) * .01)                                             SRI 096
      SC(NJ) = AO(3) - X * 100.                                         SRI 097
      FITEM(NJ) = X                                                     SRI 098
      CPPJ(NJ) = AO(21)                                                 SRI 099
      UPCJ(NJ) = AO(20)                                                 SRI 100
      WJ(NJ) = AO(24) * .1                                              SRI 101
      CJ(NJ) = AO(22) * .01 * FMCJ                                      SRI 102
      PJ(NJ) = AO(23) * .01                                             SRI 103
      SJ(NJ) = AO(13) * .0001                                           SRI 104
  203 CALL RDTP                                                         SRI 105
      IF (AO(1)) 205,204,204                                            SRI 106
  204 IF (AO(1) - 2.) 300,201,206                                       SRI 107
  205 IEOF = 2                                                          SRI 108
      GO TO 300                                                         SRI 109
  206 DO 207 K = 1,NK                                                   SRI 110
      IF (BCK(K) - AO(4)) 207,208,207                                   SRI 111
  207 CONTINUE                                                          SRI 112
      GO TO 203                                                         SRI 113
  208 BF(NJ,K) = 1.                                                     SRI 114
      DHJK(NJ,K) = AO(13)                                               SRI 115
      SIJK(NJ,K) = AO(13)                                               SRI 116
      SIJ(NJ) = SIJ(NJ) + AO(13)                                        SRI 117
      SI = SI + AO(13) * SJ(NJ)                                         SRI 118
C
C     COMPUTE MEAN RATE OF DEMAND
      X = 0.                                                            SRI 119
      DO 209 L = 1,12                                                   SRI 120
      Y = AO(L + 13)                                                    SRI 121
      X = X + Y                                                         SRI 122
  209 DM(NJ,K,L) = Y                                                    SRI 123
      DIJK(NJ,K) = X / 52.                                              SRI 124
      GO TO 203                                                         SRI 125
C                                                                       SRI 126
C     INITIALIZE ORDER QUEUES                                           SRI 127
C                                                                       SRI 128
  300 CALL QUEUEP (3)                                                   SRI 129
      CALL QUEUEB (3)                                                   SRI 130
C
C     COMPUTE OPTIMAL COVERAGE PERIODS
      GO TO (301,400),ICON1                                             SRI 131
  301 CALL RXCOMP (2)                                                   SRI 132
C                                                                       SRI 133
C     GENERATE ACTUAL DEMAND, PREDICT DEMAND, AND COMPUTE X AND R LEVELSSRI 134
C                                                                       SRI 135
  400 T = T + 1.                                                        SRI 136
      CALL PREDICT                                                      SRI 137
      CALL GENDEM                                                       SRI 138
```

```
      GO TO (401,406),ICONT                                            SRI 139
  401 QI = INTF(MINIF(FLDI, FLQI * DI ** .5) + .9)                     SRI 140
      TI = QI / DI + FLRI * .5                                         SRI 141
      RI = 0.                                                          SRI 142
      DO 404 J = 1,NJ                                                  SRI 143
      RIJK(J,1) = MAX1F(MINIF(DIJ(J) * XRIJZ(J), XDIJK(J,1)), DIJ(J) * SRI 144
     1            XXIJZ(J) + .9 + FK3 * DIJ(J) * XDIJ(J) * (TI + FL )) SRI 145
      XIJK(J,1) = DIJ(J) * XXIJZ(J)                                    SRI 146
      XIJ(J) = DIJ(J) * XXIJ(J)                                        SRI 147
      RIJ(J) = 0.                                                      SRI 148
      DO 403 K = 2,NK                                                  SRI 149
      IF  (BF(J,K)) 403,403,402                                        SRI 150
  402 XIJK(J,K) = DIJK(J,K) * XK2IK(K)                                 SRI 151
      RIJK(J,K) = MAX1F(MINIF(DIJK(J,K) * XRIJK(J,K), XDIJK(J,K)),    SRI 152
     1   XIJK(J,K) + F(1 * DIJK(J,K) * (TI + FLIK(K)))                 SRI 153
      RIJ(J) = RIJ(J) + RIJK(J,K)                                      SRI 154
  403 CONTINUE                                                         SRI 155
      RIJ(J) = RIJ(J)  + RIJK(J,1)                                     SRI 155A
  404 RI = RI + RIJ(J) * SJ(J)                                         SRI 156
      XI = RI - QI                                                     SRI 157
      IF (T - 1.) 500,405,500                                          SRI 158
  405 FLPDIM = 1. - MAX1F(1., (RI - SI) / DI)                          SRI 159
      GO TO 500                                                        SRI 160
C
C     COMPUTE R AND X LEVELS FOR ALTERNATE RULES
  406 DO 408 J = 1,NJ                                                  SRI 161
      RIJK(J,1) = DIJ(J)* PG                                           SRI 162
      DO 408 K = 2,NK                                                  SRI 163
      IF (BF(J,K)) 408,408,407                                         SRI 164
  407 RIJK(J,K) = DIJK(J,K) * BG                                       SRI 165
      XIJK(J,K) = DIJK(J,K) * BG * RGX                                 SRI 166
  408 CONTINUE                                                         SRI 167
C                                                                      SRI 168
C     FILL PRODUCTION ORDERS                                           SRI 169
C                                                                      SRI 170
  500 CALL QUEUEP(2)                                                   SRI 171
      IF (POBS) 514,514,501                                            SRI 172
  501 GO TO (502,508),ICONT                                            SRI 173
  502 X = 0.                                                           SRI 174
      DO 505 J = 1,NJ                                                  SRI 175
      OOJK(J,1) = OOJK(J,1) - POAW(J)                                  SRI 176
      SIJK(J,1) = SIJK(J,1) - POAW(J)                                  SRI 177
      SIJ(J) = SIJ(J) - POAS(J)                                        SRI 178
      SI = SI - POAS(J) * SJ(J)                                        SRI 179
      POAS(J) = MAX1F(0.,RIJK(J,1) - OHJK(J,1) -DIJ(J)*XDIJ(J)*FLP+.99) SRI 180
      DO 504 K = 2,NK                                                  SRI 181
      IF (BF(J,K)) 504,504,503                                         SRI 182
  503 POAS(J) = POAS(J) + MAX1F(0.,RIJK(J,K)-SIJK(J,K)-DIJK(J,K)*FLP+.9)SRI 183
  504 CONTINUE                                                         SRI 184
```

```
  505 X = X + POAS(J) * SJ(J)                                       SRI 185
      IF (X) 500,500,506                                            SRI 186
  506 R = POBS / X                                                  SRI 187
      DO 507 J = 1,NJ                                               SRI 188
      POAS(J) = RNDNF(POAS(J) * R, UPCJ(J))                         SRI 189
  507 OOJK(J,1) = OOJK(J,1) + POAS(J)                               SRI 190
  508 DO 509 J = 1,NJ                                               SRI 191
      OHJK(J,1) = OHJK(J,1) + POAS(J)                               SRI 192
      SIJK(J,1) = SIJK(J,1) + POAS(J)                               SRI 193
      OOJK(J,1) = OOJK(J,1) - POAS(J)                               SRI 194
      SIJ(J) = SIJ(J) + POAS(J)                                     SRI 195
  509 SI =SI + POAS(J) * SJ(J)                                      196
      GO TO (510,500),ICON1                                         SRI 197
  510 DO 512 J = 1,NJ                                               SRI198
      DO 512 K = 2,NK                                               SRI 199
      IF (BF(J,K)) 512,512,511                                      SRI 200
  511 ORD = MAX1F(0., RIJK(J,K) - DIJK(J,K) * FLP - SIJK(J,K))      SRI 201
      ORD = RNDNF (ORD, UPCJ(J))                                    SRI 202
      ODD = INTF(RNDLF(T, FSK(K)) + FSK(K) + FLSK(K) + .5)          SRI 203
      VSIT = VSIT + ORD * CJ(J)                                     SRI 204
      CALL PBO(2)                                                   SRI 205
  512 CONTINUE                                                      SRI 206
      GO TO 500                                                     SRI 207
C                                                                   SRI 208
C     FILL BRANCH ORDERS                                            SRI 209
C                                                                   SRI 210
      XFLPDTM = T - INTF(FLP + .5)
      XDD = INTF( T + FLS + FSK(2) + .9 )
  514 CALL QUEUEB(2)                                                SRI 211
      IF (ORD) 516,516,515                                          SRI 212
  515 OHJK(J,K) = OHJK(J,K) + ORD                                   SRI 213
      OOJK(J,K) = OOJK(J,K) - ORD                                   SRI 214
      VSIT = VSIT - ORD * CJ(J)                                     SRI 215
      GO TO 514                                                     SRI 216
C                                                                   SRI 217
C     FILL DEMANDS ON BRANCHES                                      SRI 218
C                                                                   SRI 219
  516 DO 524 J = 1,NJ                                               SRI 220
      DO 524 K = 1,NK                                               SRI 221
      IF (BF(J,K)) 524,524,517                                      SRI 222
  517 IF (DWIJK(J,K)) 524,524,518                                   SRI 223
  518 ZDIJK(J,K) = ZDIJK(J,K) + DWIJK(J,K)                          SRI 224
      IF (OHJK(J,K)) 522,522,519                                    SRI 225
  519 TP = OHJK(J,K) - DWIJK(J,K)                                   SRI 226
      TP2 = DWIJK(J,K)                                              SRI 227
      IF (TP) 520,520,523                                           SRI 228
  520 GO TO (521,522),ICON3                                         SRI 229
  521 TP2 = OHJK(J,K)                                               SRI 230
```

```
      TP = 0.                                                           SRI 231
      SOCJK(J,K) = SOCJK(J,K) + 1.                                      SRI 232
      GO TO 523                                                         SRI 233
  522 TP = CHJK(J,K)                                                    SRI 234
      TP2 = 0.                                                          SRI 235
  523 OHJK(J,K) = TP                                                    SRI 236
      DSIJK(J,K) = DSIJK(J,K) + TP2                                     SRI 237
      SIJK(J,K) = SIJK(J,K) - TP2                                       SRI 238
      SIJ(J) = SIJ(J) - TP2                                             SRI 239
      SI = SI - TP2 * SJ(J)                                             SRI 240
  524 CONTINUE                                                          SRI 241
      GO TO (600,1600),ICON1                                            SRI 242
C                                                                       SRI 243
C     SRI BRANCH ORDERING PROCEEDURE                                    SRI 244
C                                                                       SRI 245
      IF(T - XDD ) 600,580,600
  580 FLPDTM = XFLPDTM
  600 DO 608 J = 1,NJ                                                   SRI 246
      DO 608 K = 2,NK                                                   SRI 247
      IF (BF(J,K)) 608,608,601                                          SRI 248
  601 IF (SIJK(J,K) - XIJK(J,K)) 602,602,608                            SRI 249
  602 TI2  = T - FLPDTM                                                 SRI 250
      ORD  = INTF(MAXIF(XIJK(J,K) + DIJK(J,K)* MINIF(TI , FLIK(K)) *.5, SRI 251
     1           RIJK(J,K) - DIJK(J,K)*TI2) - SIJK(J,K) + .99)          SRI 252
      GO TO(604,603),ICON4                                              SRI 253
  603 ORD  = RNDCF (ORD, UPCJ(J) )                                      SRI 254
  604 GO TO (607,605),ICON5                                             SRI 255
  605 Y =  UPCJ(J) * CPPJ(J)                                            SRI 256
      PC = (ORD  - INTF(ORD /Y )* Y ) * CBPC                            SRI 257
      X  =  RNDCF (ORD , Y)                                             SRI 258
      HC = (X - ORD )* CPPJ(J) * XRIJK(J,K) * .325 / 52.                SRI 259
      IF  (HC - PC) 606,607, 607                                        SRI 260
  606 ORD = X                                                           SRI 261
  607 ODD = INTF(RNDLF(T, FSK(K)) + FSK(K) + FLSK(K) + .5)              SRI 262
      VSIT = VSIT + ORD * CJ(J)                                         SRI 263
      CALL  PBO (1)                                                     SRI 264
  608 CONTINUE                                                          SRI 265
C                                                                       SRI 266
C     SRI PRODUCTION ORDERING PROCEDURE                                 SRI 267
C                                                                       SRI 268
      IF (SI - XI) 613,613,610                                          SRI 269
  610 DO 612 J = 1,NJ                                                   SRI 270
      IF (SIJK(J,1) - XIJK(J,1) ) 613,613,611                           SRI 271
  611 IF (SIJ(J) - XIJ(J) ) 613,613,612                                 SRI 272
  612 CONTINUE                                                          SRI 273
      GO TO 700                                                         SRI 274
  613 POBS = 0.                                                         SRI 275
```

```
      DO 616 J = 1,NJ                                                   SRI 276
      POAW(J) = MAX1F(0., RIJK(J,1) - SIJK(J,1))                        SRI 277
      POAS(J) = POAW(J)                                                 SRI 278
      DO 615 K = 2,NK                                                   SRI 279
      IF (BF(J,K) ) 615,615, 614                                        SRI 280
  614 POAS(J) = POAS(J) + MAX1F (0., RIJK(J,K) - SIJK(J,K ))            SRI 281
  615 CONTINUE                                                          SRI 282
  616 POBS = POBS + POAS(J) * SJ(J)                                     SRI 283
      IF (POBS) 700,700,617                                             SRI 284
  617 ORD = POBS                                                        SRI 285
      POBS = RNDCF ( POBS, BSI )                                        SRI 286
      GO TO 650                                                         SRI 287
C                                                                       SRI 288
C     ALTERNATE BRANCH ORDERING PROCEDURE
C                                                                       SRI 290
 1600 X = T - INTF(T / FLRI) * FLRI                                     SRI 291
      IF (X) 700,1601,700                                               SRI 292
 1601 DO 1606 J = 1,NJ                                                  SRI 293
      R = 1.                                                            SRI 294
      IF (XIJK(J,1) - SIJK(J,1)) 1603,1603,1602                         SRI 295
 1602 R = .5                                                            SRI 296
 1603 DO 1606 K = 2,NK                                                  SRI 297
      IF (BF(J,K)) 1605,1606,1604                                       SRI 298
 1604 IF (XIJK(J,K) - SIJK(J,K)) 1606,1605,1605                         SRI 299
 1605 ORD = (RIJK(J,K) - SIJK(J,K)) * R                                 SRI 300
      ODD = INTF(RNDLF(T, FSK(K)) + FSK(K) + FLSK(K) + .5)              SRI 301
      VSIT = VSIT + ORD * CJ(J)                                         SRI 302
      CALL PBO (1)                                                      SRI 303
 1606 CONTINUE                                                          SRI 304
C                                                                       SRI 305
C     ALTERNATE PRODUCTION ORDERING PROCEDURE
C                                                                       SRI 307
      POBS = 0.                                                         SRI 308
      DO 1607 J = 1,NJ                                                  SRI 309
      POAS(J) = MAX1F(0., RIJK(J,1) - SIJK(J,1))                        SRI 310
      POAW(J) = POAS(J)                                                 SRI 311
 1607 POBS = POBS + POAS(J) * SJ(J)                                     SRI 312
      IF (POBS) 700,700,1608                                            SRI 313
 1608 ORD = POBS                                                        SRI 314
      POBS = MAX1F(BSM, RNDCF(POBS,BSI))                                SRI 315
C                                                                       SRI 316
C     COMMON PRODUCTION ORDERING SECTION                                SRI 317
C                                                                       SRI 318
  650 R = POBS / ORD                                                    SRI 319
      PODD = T + INTF(FLP + .5)                                         SRI 320
      DO 651 J = 1,NJ                                                   SRI 321
      POAS(J) = RNDNF(POAS(J) * R, UPCJ(J))                             SRI 322
      POAW(J) = RNDNF(POAW(J) * R, UPCJ(J))                             SRI 323
      OOJK(J,1)  = OOJK(J,1)  + POAW(J)                                 SRI 324
```

```
      SIJK(J,1)  = SIJK(J,1)  + POAW(J)
      SIJ(J) = SIJ(J) + POAS(J)
  651 SI = SI + POAS(J) * SJ(J)
      FNOPR = FNOPR + 1.
      CALL QUEUEP (1)
C
C     END-OF-WEEK HOUSEKEEPING
C
  700 DO 703 J = 1,NJ
      DO 703 K = 1,NK
      IF (BF(J,K)) 703,703,701
  701 SYJK(J,K) = SYJK(J,K) + OHJK(J,K) / 52.
      IF (OHJK(J,K)) 702,702,703
  702 SOWJK(J,K) = SOWJK(J,K) + 1.
  703 CONTINUE
C     ***** THE FOLLOWING STATEMENTS ARE FOR CHECKOUT ONLY *****
      PRINT 91, T, SI, SIJ(1), SIJ(2)
      DO 705 J = 1,NJ
      DO 705 K = 1,NK
      IF (BF(J,K)) 705,705,704
  704 PRINT 92, J, K, SIJK(J,K), OHJK(J,K), COJK(J,K)
  705 CONTINUE
      VYSIT = VYSIT + VSIT / 52.
      IF (T - 52.) 400,800,800
C
C     PRINT ITEM SUMMARY
C
  800 PAGE = PAGE +1.
      DO 86 J = 1,NJ
      DO 86 K = 1,NK
      IF (BF(J,K)) 86,86,85
   85 PRINT 84, J, K, DWIJK(J,K), RIJK(J,K), XIJK(J,K), XRIJK(J,K),
     1          XDIJK(J,K), XDIJ(J), XXIJ(J), XRIJZ(J), XXIJZ(J),
     2          XK2IK(K), SIJK(J,K)
   86 CONTINUE
      PRINT 10, RUNNO, PAGE, BASE
      VSYI  = 0.
      CHI   = 0.
      VDSI  = 0.
      DCI   = 0.
      SOWI  = 0.
      SOCI  = 0.
      VZDI  = 0.
      CLI   = 0.
      BCPI  = 0.
      CBCI  = 0.
      CBPI  = 0.
```

```
      DO 806 J = 1,NJ
      BCPCJ = 0.
      VSYJ = 0.
      SYJ  = 0.
      CHJ  = 0.
      OCJ  = 0.
      VDSJ = 0.
      DSIJ = 0.
      SOWJ = 0.
      SOCJ = 0.
      ULJ  = 0.
      CLJ  = 0.
      ZDIJ = 0.
      VZDJ = 0.
      CBCJ = 0.
      CBPJ = 0.
      DO 805 K = 1,NK
      IF (BF(J,K)) 805,805,801
  801 VSYJK = SYJK(J,K) * CJ(J)
      VSYK(K) = VSYK(K) + VSYJK
      VSYJ = VSYJ + VSYJK
      SYJ = SYJ + SYJK(J,K)
      CHJK = VSYJK * AZR
      IF (K - 1) 803,802,803
  802 CHJK = CHJK + SYJK(J,K) * SJ(J) * APP
      GO TO 804
  803 CHJK = CHJK + SYJK(J,K) * SJ(J) * AP3
  804 CHK(K) = CHK(K) + CHJK
      CHJ = CHJ + CHJK
      OCJ = OCJ + OCJK(J,K)
      VDSJK = DSIJK(J,K) * CJ(J)
      VDSK(K) = VDSK(K) + VDSJK
      VDSJ = VDSJ + VDSJK
      DSIJ = DSIJ + DSIJK(J,K)
      RSTJK = DSIJK(J,K) / SYJK(J,K)
      SOWJ = SOWJ + SOWJK(J,K)
      SOWK(K) = SOWK(K) + SOWJK(J,K)
      SOCJ = SOCJ + SOCJK(J,K)
      WPSOJK = SOWJK(J,K) / SOCJK(J,K)
      ULJK = ZDIJK(J,K) - DSIJK(J,K)
      CLJK = ULJK * (PJ(J) - CJ(J))
      CLK(K) = CLK(K) + CLJK
      ULJ = ULJ + ULJK
      CLJ = CLJ + CLJK
      ZDIJ = ZDIJ + ZDIJK(J,K)
      VZDJ = ZDIJ * CJ(J)
      VZDK(K) = VZDK(K) + ZDIJK(J,K) * CJ(J)
      RLTJK = ULJK / ZDIJK(J,K)
```

```
      BCPCJK = BCCJK(J,K) + BPCJK(J,K)                                  SRI 419
      CBCJK  = BCCJK(J,K) * CBCC                                        SRI 420
      CBPJK  = BPCJK(J,K) * CBPC                                        SRI 421
      BCPCJ = BCPCJ + BCPCJK                                            SRI 422
      CBCJ = CBCJ + CBCJK                                               SRI 423
      CBPJ = CBPJ + CBPJK                                               SRI 424
      CPLK(K) = CPLK(K) + CBCJK + CBPJK                                 SRI 425
C
C     PRINT ITEM-LOCATION DATA
      PRINT 11,FITEM(J),SC(J),RCK(K),SYJK(J,K),VSYJK,CHJK,DSIJK(J,K),   SRI 426
     1   VDSJK,RSTJK,OCJK(J,K),SOWJK(J,K),SOCJK(J,K),WPSOJK,ULJK,CLJK,  SRI 427
     2         RLTJK,BCPCJK,CBCJK,CBPJK                                 SRI 428
  805 CONTINUE                                                          SRI 429
      RSTJ = DSIJ / SYJ                                                 SRI 430
      WPSOJ = SOWJ / SOCJ                                               SRI 431
      RLTJ = ULJ / ZDIJ                                                 SRI 432
      VSYI = VSYI + VSYJ                                                SRI 433
      CHI = CHI + CHJ                                                   SRI 434
      VDSI = VDSI + VDSJ                                                SRI 435
      OCI = OCI + OCJ                                                   SRI 436
      SOWI = SOWI + SOWJ                                                SRI 437
      SOCI = SOCI + SOCJ                                                SRI 438
      CLI = CLI + CLJ                                                   SRI 439
      VZDI = VZDI + VZDJ                                                SRI 440
      BCPI = BCPI + BCPCJ                                               SRI 441
      CBCI = CBCI + CBCJ                                                SRI 442
      CBPI = CBPI + CBPJ                                                SRI 443
C
C     PRINT ITEM TOTALS
  806 PRINT 12,SYJ,VSYJ,CHJ,DSIJ,VDSJ,RSTJ,OCJ,SOWJ,SOCJ,WPSOJ,ULJ,CLJ, SRI 444
     1         RLTJ,BCPCJ,CBCJ,CBPJ                                     SRI 445
      RSTI = VDSI / VSYI                                                SRI 446
      WPSOI = SOWI / SOCI                                               SRI 447
      RLTI = CLI / VZDI                                                 SRI 448
C
C     PRINT PRODUCT TOTALS
      PRINT 13,VSYI,CHI,VDSI,RSTI,OCI,SOWI,SOCI,WPSOI,CLI,RLTI,         SRI 449
     1         BCPI,CBCI,CBPI                                           SRI 450
      VSY = VSY + VSYI                                                  SRI 451
      CH = CH + CHI                                                     SRI 452
      VDS = VDS + VDSI                                                  SRI 453
      SY = SY + SYI                                                     SRI 454
      OC = OC + OCI                                                     SRI 455
      SOW = SOW + SOWI                                                  SRI 456
      SOC = SOC + SOCI                                                  SRI 457
      CL = CL + CLI                                                     SRI 458
      VZD = VZD + VZDI                                                  SRI 459
      UL = UL + ULI                                                     SRI 460
```

```
      BCP = BCP + BCPI
      CBC = CBC + CBCI
      CBP = CBP + CBPI
      GO TO (200,900),IEOF
C
C     PRINT SYSTEM TOTALS
C
  900 PAGE = PAGE + 1.
      RST = VDS / VSY
      WPSO = SOW / SOC
      RLT = CL / VZD
      PRINT 14, VSY,CH,VDS,RST,OC,SOW,SOC,WPSO,CL,RLT,BCP,CBC,CBP,
     1          RUNNO, PAGE
      VZD = 0.
      VDS = 0.
      VSY = 0.
      CH  = 0.
      OC  = 0.
      CPP = 0.
      CPL = 0.
      CF  = 0.
      CP  = 0.
      CBO = 0.
      SOW = 0.
      CL  = 0.
      ZVC = 0.
      DO 902 K = 1,NK
      IF (BCK(K)) 902,902,901
  901 OCK = 52. / FSK(K)
      CPPK = OCK * CPPJ
      RSTK = VDSK(K) / VSYK(K)
      CBOK = CPPK + CPLK(K) + CFK(K) + CPK(K)
      ZVCK = CHK(K) + CBOK + CLK(K)
      CPDK = CLK(K) / VZDK(K)
      VDS = VDS + VDSK(K)
      VSY = VSY + VSYK(K)
      CH  = CH  + CHK(K)
      OC  = OC  + OCK
      CPP = CPP + CPPK
      CPL = CPL + CPLK(K)
      CF  = CF  + CFK(K)
      CP  = CP  + CPK(K)
      CBO = CBO + CBOK
      SOW = SOW + SOWK(K)
      CL  = CL  + CLK(K)
      ZVC = ZVC + ZVCK
      VZD = VZD + VZDK(K)
```

```
C
C     PRINT BRANCH SUMMARY REPORT AND STOP
      PRINT 15,BCK(K),VDSK(K),VSYK(K),CHK(K),RSTK,OCK,CPPK,CPLK(K),     SRI 508
     1         CFK(K),CPK(K),CBOK,SOWK(K),CLK(K),CPDK,ZVCK              SRI 509
  902 CONTINUE                                                          SRI 510
      RST = VDS / VSY                                                   SRI 511
      CPD = CL / VZD                                                    SRI 512
      PRINT 16,VDS,VSY,CH,RST,OC,CPP,CPL,CF,CP,CBO,SOW,CL,CPD,ZBC       SRI 513
      CP = A * FNOPR                                                    SRI 514
      CTOT = CPL + CP + CL + CH                                         SRI 515
      PAGE = PAGE + 1.                                                  SRI 516
      PRINT 18, RUNNO, PAGE                                             SRI 517
      GO TO (903,904),ICON1                                             SRI 518
  903 CALL RXCOMP (3)                                                   SRI 519
  904 PRINT 17, CPL, CP, CL, CH, CTOT, VYSIT                            SRI 520
 1000 STOP                                                              SRI 521
C                                                                       SRI 522
C     FORMATS                                                           SRI 523
C                                                                       SRI 524
    1 FORMAT (8X,A6,10X5I2,6X,3F4.2,5F5.2/11X6F4.2,20X,2F3.0,3X,F3.0,5X,SRI 525
     1          F3.0/3XF5.3,16X2F3.1,8X3F3.1,12XF4.2,F8.2//25F3.0/      SRI 526
     2          25F3.2 / 25F3.0 / 25F3.1 / )                            SRI 527
    2 FORMAT (1H0,10X,22HBASE RULES COMPUTATION,           /1H0,10X,3HSRI 528
     1AZR,5X,F7.3,11X,3HAPB,F12.3,11X,3HAP2,F12.3,11X,2HCR,F13.3/11X,2HLSRI 529
     2R,F13.3,11X,2HLC,F13.3,11X,2HLS,F13.3,11X,2HA1,F13.3/11X,2HA3,F13.SRI 530
     33,11X,2HMU,F13.3,11X,2HK6,F13.3,11X,2HK5,F13.3/11X,2HK1,F13.3,11X,SRI 531
     42HK2,F13.3,11X,2HK3,F13.3,11X,3HCM2,F12.3/11X,4HVLP1,F11.3,11X,4HVSRI 532
     5LP3,F11.3/1H0,10X,11HBRANCH DATA/1H0,10X5HCODE ,25F4.0/1H010X,2HCSSRI 533
     6,3X,25F4.2/1H0,10X1HW,4X,25F4.0/1H0,10X,3HMUK,2X,25F4.1/1H0/1H0)  SRI 534
    3 FORMAT (37H1CONTROL CARD INPUTS  --  RUN NUMBER A6/1H05X13HSELECTOSRI 535
     1RS  --7X4HRULEI41/26X8HSTOCKOUTI37/26X14HROUND TO CASESI31/26X15HRSRI 536
     2OUND TO PALLETI33/1H05X9HCOSTS  --11X11HOPPORTUNITYF34.3/26X27HPHYSRI 537
     3SICAL STORAGE - BRANCHESF18.2/43X11H- WAREHOUSEF17.2/26X15HORDER PSRI 538
     4REPARINGF30.2/26X36HPICKING AND LOADING - BROKEN PALLETSF9.2/46X14SRI 539
     5H- BROKEN CASESF11.2/46X14H- FULL PALLETSF11.2/26X9HRECEIVINGF36.2SRI 540
     6/1H05X15HMULTIPLIERS  --5X15HDEMAND FORECASTF30.2/26X24HLEAD TIME SRI 541
     7-WHS TO BRANCHF21.2/36X14H- PLANT TO WHSF21.2/26X14HGOALS - BRANCHSRI 542
     8F31.2/32X5H- WHSF34.2/26X13HSTANDARD COSTF32.2/1H05X12HLEAD TIME -SRI 543
     9-8X8HREACTIONF37.0/26X13HCOMMUNICATIONF32.0/26X10HIN TRANSITF35.0)SRI 544
   10 FORMAT(1H120X9HRUN NO. A6,4X9HPAGE NO. F4.0,5X4HBASEF9.0/1H065X,  SRI 545
     1 33H----------- STOCKOUTS ----------- /93X,6HRATIO /              SRI 546
     2 18X,40H- AVERAGE INVENTORY -    TOTAL      STOCK,17X,            SRI 547
     3 42HWEEKS ESTD           LOST      BROKEN CARTON- /10H - BASE --,21X,SRI 548
     4 33HHOLDING  -- SALES --   TURN   ORDER,12X,                      SRI 549
     5 43HPER  UNITS ESTD  DEMAND       PALLET COSTS   /119H ITEM SIZE BRASRI 550
     6NCH UNITS   VALUE    COST   UNITS VALUE  RATIO COUNT WEEKS FREQ S/O SRI 551
     7 LOST  COST  IO TOT     NO. CTNS    PLTS  / )                     SRI 552
   11 FORMAT (1H0,F5.0,F3.0,F6.0,F7.0,F8.0,2F7.0,F8.0,F6.1,F5.0,F6.0,   SRI 553
```

```
      1         3F5.0,F7.0,F7.2,F8.0,F6.0,F7.0)                                SRI 554
   12 FORMAT (1HO,4X,10HSIZE TOTAL,F7.0,F8.0,2F7.0,F8.0,F6.1,F5.0,F6.0,        SRI 555
      1         3F5.0,F7.0,F7.2,F8.0,F6.0,F7.0/1HO)                            SRI 556
   13 FORMAT (1HO,3X,11H BASE TOTAL,F15.0,F7.0,F15.0,F6.1,F5.0,F6.0,           SRI 557
      1         2F5.0,F12.0,F7.2,F8.0,F6.0,F7.0)                               SRI 558
   14 FORMAT (15HO* SYSTEM TOTAL F15.0,F7.0,F15.0,F6.1,F5.0,F6.0,F5.0,         SRI 559
      1         F5.0,F12.0,F7.2,F8.0,F6.0,F7.0 /                               SRI 560
      2         1H1,20X,3HRUN NO. A6,4X,8HPAGE NO.,F5.0/                       SRI 561
      2         1HO,16X,20H- INVENTORY VALUES -  8X                            SRI 562
      3         ,66H----- VARIABLE COSTS OF BRANCH ORDERS ------    ---- STSRI 563
      3OCKOUTS ----- / 114X,5HTOTAL / 26X,35HHOLD. STK   ORDER  PREP.   PSRI 564
      4ICKING-,10X,10HUNLOADING-,10X,29HITEM-  EST  COST PER VARIABLE /  SRI 565
      583H BRANCH  DEMAND  AVERAGE  COST  TURN  COUNT  PROC.   LOADING  FSRI 566
      6REIGHT  PROCESSING  ,36HTOTAL    WEEKS  COST $ DEMAND   COSTS )   SRI 567
   15 FORMAT (1HO,F4.0,F10.0,F8.0,F8.0,F6.1,F5.0,F9.0,F8.0,F9.0,F11.0,         SRI 568
      1         F10.0,F6.0,F8.0,F7.2,F9.0)                                     SRI 569
   16 FORMAT (1HO,5HTOTAL,F9.0,F8.0,F8.0,F5.1,F5.0,F9.0,F8.0,F9.0,F11.0,SRI 570
      1         F10.0,F6.0,F8.0,F7.2,F9.0)                                     SRI 571
   17 FORMAT (1HO,44X,19HBRANCH PICKING COSTF13.0/45X,15HPRODUCTION COSTSRI 572
      1         F17.0/45X,13HSHORTAGE COST F19.0/45X,12HHOLDING COST F20.0SRI 573
      2         /1HO,44X,12H* TOTAL COST F20.0/                                SRI 574
      1         1HO,44X,24HAVERAGE VALUE IN TRANSIT F8.0)                      SRI 575
   18 FORMAT (1H1,9X,13HBASE RULES SUMMARY,20X,8HRUN NO.  ,A6,5X,              SRI 576
      1         8HPAGE NO.,F5.0)                                               SRI 577
   84 FORMAT (2HOA, 2I4,  11F9.1)                                              SRI 578
   91 FORMAT (2HOS 4F10.1)                                                     SRI 579
   92 FORMAT (2H  2I5, 3F10.0)                                                 SRI 580
      END                                                                      SRI 581
C
      SUBROUTINE RXCOMP (ICON)                                                 SRI 582
C                                                                              SRI 583
C     LOCAL STORAGE                                                            SRI 584
C                                                                              SRI 585
      DIMENSION CSZ(20), CHJJ(20), CIJ(25), FLZK(25), VLIK(25),                SRI 586
     1          VDPR(20), XJ1(20), XVDI(20), XCI(20), DPR(20)                  SRI 587
C                                                                              SRI 588
C     COMMON STORAGE                                                           SRI 589
C                                                                              SRI 590
      DIMENSION AO(26), BCCJK(20,25), BF(20,25), BPCJK(20,25),                 SRI 591
     1          CFK(25), CJ(20), CPK(25), CPLK(25), CPPJ(20), DIJ(20),         SRI 592
     2          DIJK(20,25), DM(20,25,12), DWIJK(20,25), FLIK(25),             SRI 593
     3          FLSK(25), FMUK(25), FSK(25), FRK(25), OCJK(20,25),             SRI 594
     4          OHJK(20,25), OOJK(20,25), PJ(20), POAS(20),                    SRI 595
     5          POAW(20), SIJK(20,25), SJ(20), UPCJ(20), WJ(20),               SRI 596
     6          XDIJ(20), XDIJK(20,25), XK2IK(20), XRIJK(20,25),               SRI 597
     7          XRIJZ(20), XXIJ(20), XXIJZ(20), XIJ(20)                        SRI 598
      COMMON    AO, A, APB, APP, AZR, BCCJK, BF, BG, BPCJK, CFK, CJ,           SRI 599
     1          CM2, CPK, CPLK, CPPJ, CR, DI, DIJ, DIJK, DM, DWIJK,            SRI 600
```

```
     2          FLIK, FLSK, FMU, FMUK, FLP, FLRI, FK1, FK2, FK3, FLC,
     3          FK4, FK5, FK6, FSK, FRK, J, K, NJ, NK, OCJK, ODD,
     4          OHJK, OOJK, ORD, PG, PJ, POBS, PODD, POAS, POAW, QI,
     5          SIJK, SJ, T, TX, UPCJ, WJ, XDIJ, XDIJK, XK2IK, XIJ,
     6          XRIJK, XRIJZ, XXIJ, XXIJZ
C
C     FUNCTION DEFINITION
C
      VARDF(D,C) = (52. * D) ** (10.3612 / (17.269 + LOGF(C))) / D
C
      GO TO (1, 2, 3),ICON
C
C     INITIALIZE
C
    1 SD   = 0.
      SDB  = 0.
      SGP  = 0.
      SGPP = 0.
      SDLB = 0.
      SDL  = 0.
      SRP  = 0.
      SORD = 0.
      SQ   = 0.
      SQP  = 0.
      SR   = 0.
      RETURN
C
C     COMPUTE BASE RULES
C
C     COMPUTE F, DI, DICHI, DIPR
C
    2 F = 0.
      DI = 0.
      DICHI = 0.
      DIPR = 0.
      FL = FLRI / 2. + FLC + FLP
      DO 302 K = 1,NK
      FLIK(K) = FL + FLSK(K) + FSK(K) * .5
      FLZK(K) = FLRI / 2. + FLSK(K) + FSK(K) * .5
      XK2IK(K) = FLZK(K) * FK2
      T5 = FSK(K) * FSK(K) + FLRI * FLRI
      VLIK(K) = (FLP * FLP * VLP + T5 / 12.) / (FLIK(K) * FLIK(K))
      T1 = 1. + VLIK(K)
      DO 302 J = 1,NJ
      IF (BF(J,K)) 302,302,301
  301 CIJ(K) = CJ(J) + .01 * (FRK(K) + CR) * WJ(J)
      VDIJK = VARDF (DIJK(J,K), CIJ(K))
      DIPR = DIPR + 1. / VDIJK
```

```
      T2 = T1 + VDIJK / FLIK(K)                                         SRI 649
      FLKIJK = (T1 + T2 - T1 / T2) / 3.                                 SRI 650
      FIJK = FLKIJK * DIJK(J,K) * FLIK(K)                               SRI 651
      F = F + SJ(J) * FIJK                                              SRI 652
      CHJK = (AZR * CIJ(K) + APB * SJ(J)) / 52.                         SRI 653
      DICHI = DICHI + CHJK * DIJK(J,K)                                  SRI 654
      DIJ(J) = DIJ(J) + DIJK(J,K)                                       SRI 655
      DI = DI + DIJK(J,K) * SJ(J)                                       SRI 656
  302 CONTINUE                                                          SRI 657
C     ****** THE FOLLOWING STATEMENT IS FOR CHECKOUT ONLY *********     SRI 658
      PRINT 50, DI, F, DICHI, DIPR, TX, FL?, BG, PG, FK5, A             SRI 659
C                                                                       SRI 660
C     COMPUTE Q, Q-PRIME, AND  T                                        SRI 661
C                                                                       SRI 662
      F2 = F * F                                                        SRI 663
      CHI = DICHI / DI                                                  SRI 664
      SQWI2 = 2. * A * CM2 * DI / (CHI * F2 * EXPF(FMU))                SRI 665
      IF (SQWI2 - 2.25) 332,331,331                                     SRI 666
  331 SQI = SQRTF(SQWI2) + 1.                                           SRI 667
      GO TO 333                                                         SRI 668
  332 SQI = 1.85 * SQWI2 ** .3333                                       SRI 669
  333 QIZ = F * SQI                                                     SRI 670
      FLDI = DI * (TX - 1.5 * FL)                                       SRI 671
      QIZZ = MINIF(QIZ, FLDI)                                           SRI 672
      QI = INTF(QIZZ + .9)                                              SRI 673
      DSCAJ = (FLRI + 1. / DIPR) / 2.                                   SRI 674
      STI = QI / DI + DSCAJ                                             SRI 675
      QIPR = STI * DIPR                                                 SRI 676
C     ****** THE FOLLOWING STATEMENT IS FOR CHECKOUT ONLY *********     SRI 677
      PRINT 51, SQI, F , QIZ, STI, DSCAJ                                SRI 678
      FLQI = QIZ / SQRTF(DI)                                            SRI 679
C
C     COMPUTE OPTIMAL R AND X LEVELS FOR BRANCHES
      FK4 = FK1                                                         SRI 680
      RI = 0.                                                           SRI 681
      DZ = 0.                                                           SRI 682
      DBI = 0.                                                          SRI 683
      DLBI = 0.                                                         SRI 684
      DIPRPR = 0.                                                       SRI 685
      DO 349 J = 1,NJ                                                   SRI 686
      DLBJ = 0.                                                         SRI 687
      RIJJ = 0.                                                         SRI 688
      DBIJJ = 0.                                                        SRI 689
      CSZ(J) = 0.                                                       SRI 690
      DPR(J) = 0.                                                       SRI 691
      VDPR(J) = 0.                                                      SRI 692
      DO 340 K = 2,NK                                                   SRI 693
      IF (BF(J,K)) 340,340,338                                          SRI 694
```

```
338 CIJ(K) = CJ(J) + .01 * (FRK(K) + CR) * WJ(J)                        SRI 695
    VDIJK = VARDF (DIJK(J,K), CIJ(K))                                    SRI 696
    T1 = 1. + VLIK(K)                                                    SRI 697
    T2 = T1 + VDIJK / FLIK(K)                                            SRI 698
    FLKIJK = (T1 + T2 - T1 / T2) / 3.                                    SRI 699
    FIJK = FLKIJK * DIJK(J,K) * FLIK(K)                                  SRI 700
    CHJK = (AZR * CIJ(K) + APB * SJ(J)) / 52.                            SRI 701
    FKIJK = FLIK(K) * T2 / (2. * FLKIJK)                                 SRI 702
    CSIJK = PJ(J) - CIJ(K)                                               SRI 703
    BPIJK = (EXPF(DIJK(J,K) * VDIJK / FIJK) - 1.) / (DIPR * VDIJK) +1.SRI 704
    RPJK = FIJK * LOGF(FKIJK * (BPIJK ** QIPR - 1.) * (1. + CSIJK /      SRI 705
   1     (CHJK * FLKIJK * FLIK(K))) / STI)                               SRI 706
    RZJK = RPJK - (FMU + FMUK(K)) * FIJK                                 SRI 707
    RZZJK = MIN1F(RZJK, DIJK(J,K) * TX)                                  SRI 708
    XJK = INTF(FK2 * DIJK(J,K) * FLZK(K) + .99)                          SRI 719
    RJK = INTF(MAX1F(RZZJK, XJK + FK4 * DIJK(J,K) * (STI + FLIK(K)))     SRI 720
   1    + .99)                                                           SRI 721
    RIJJ = RIJJ + RJK                                                    SRI 722
    XRIJK(J,K) = MAX1F(0., RZJK / DIJK(J,K))                             SRI 732
    XDIJK(J,K) = DIJK(J,K) * TX                                          SRI 738
C   ****** THE FOLLOWING STATEMENT IS FOR CHECKOUT ONLY *********        SRI 709
    PRINT 54, DIJK(J,K), VDIJK, FIJK, DIPR, BPIJK, FLKIJK, QIPR,         SRI 710
   1   CSIJK, CHJK, FLIK(K), STI, RPJK, FMU, FMUK(K), RZJK, RZZJK, TX SRI 711
C
C   COMPUTE EXPECTED VALUE OF INTERIM BRANCH ORDERS
    STIK = STI + FLIK(K)                                                 SRI 723
    DZIJK = DIJK(J,K) * SQRTF(((STIK + VDIJK) * STIK + VLIK(K) *         SRI 724
   1     FLIK(K) ** 2 + STI / DIPR) / 2.)                                SRI 725
    PRIJK = EXPF((XJK - RJK) / DZIJK)                                    SRI 726
    QRIJK = RJK - XJK + (DSCAJ - STI - FLIK(K) / 2.) * DIJK(J,K)         SRI 727
    QRIJK = MAX1F(QRIJK, DIJK(J,K) * FLIK(K) / 2., DIJK(J,K) * STI/2.)SRI 728
    DPR(J) = DPR(J) + PRIJK * QRIJK                                      SRI 729
C
C   COMPUTE EXPECTED COST OF SHORTAGES AT WAREHOUSE
    T5 = FSK(K) * FSK(K) + FLRI * FLRI                                   SRI 712
    VLZK = (T5 / FLZK(K) ** 2) *.083333                                  SRI 713
    T1 = 1. + VLZK                                                       SRI 714
    T2 = T1 + VDIJK / FLZK(K)                                            SRI 715
    FKZJK = (T1 + T2 - T1 / T2) / 3.                                     SRI 716
    FZJK = FKZJK * DIJK(J,K) * FLZK(K)                                   SRI 717
    CKZJK = FLZK(K) * T2 / (2. * FKZJK)                                  SRI 718
    VDZJK = STI / PRIJK                                                  SRI 730
    VDPR(J) = VDPR(J) + PRIJK                                            SRI 731
    T2 = XJK / FZJK                                                      SRI 735
    CSZ(J) = CSZ(J) + (PJ(J) - CIJ(K)) * ((1. + T2) / FLZK(K) * EXPF( SRI 736
   1     1. - T2) * DIJK(J,K) * CKZJK)/ (VDZJK * QRIJK)                  SRI 737
    DBIJJ = DBIJJ + DIJK(J,K)                                            SRI 733
    DLBJ = DLBJ + FLIK(K) * DIJK(J,K)                                    SRI 734
```

```
C      ****** THE FOLLOWING STATEMENT IS FOR CHECKOUT ONLY *********      SRI 739
       PRINT 52, FLIK(K), XK2IK(K), XDIJK(J,K), QRIJK, XJK, RJK, CSIJK    SRI 740
   340 CONTINUE                                                           SRI 741
       DIJ(J) = DBIJJ + DIJK(J,1)                                         SRI 746
       DLIJ = DLBJ + FLIK(1) * DIJK(J,1)                                  SRI 747
       DLI = DLI + SJ(J) * DLIJ                                           SRI 748
       DLBI = DLBI + SJ(J) * DLBJ                                         SRI 749
       DBI = DBI + SJ(J) * DBIJJ                                          SRI 750
       DZ = DZ + DIJK(J,1) * SJ(J)                                        SRI 752
C
C      COMPUTE ITEM AND WAREHOUSE REORDER LEVELS
       DPR(J) = DPR(J) / STI + DIJK(J,1)                                  SRI 742
       XRIJK(J,1) = 0.                                                    SRI 743
       XDIJK = DPR(J) * TX                                                SRI 744
       XK2IK(1) = FK2 * FLZK(1)                                           SRI 745
       XIJ(J) = FK5 * DLIJ                                                SRI 751
       XXIJ(J) = XIJ(J) / DIJ(J)                                          SRI 753
       XDIJ(J) = DPR(J) / DIJ(J)                                          SRI 754
       XJ1(J) = FK6 * DPR(J) * FL                                         SRI 755
       XXIJZ(J) = XJ1(J) / DIJ(J)                                         SRI 756
C
C      COMPUTE WAREHOUSE REQN LEVEL AND PRODUCT REORDER LEVEL
       RI = RI + SJ(J) * RIJJ                                             SRI 757
       CIJ1 = CJ(J) + .01 * (FRK(1) + CR) * WJ(J)                         SRI 758
       XCI(J) = CIJ1                                                      SRI 759
       VDIJ1 = VARDF (DIJK(J,1), CIJ1)                                    SRI 760
       XVDI(J) = VDIJ1                                                    SRI 761
       CHJJ(J) = (AZR * CIJ1 + APP * SJ(J)) / 52.                         SRI 762
       VDPR(J) = 1. / (VDPR(J) / STI + 1. / VDIJ1)                        SRI 763
       DIPRPR = DIPRPR + 1. / VDPR(J)                                     SRI 764
   349 CONTINUE                                                           SRI 767
   350 FK4 = FK3                                                          SRI 768
       DIPR = DIPR + DIPRPR                                               SRI 769
       RBI = RI                                                           SRI 770
       QIPR = STI * DIPR                                                  SRI 771
       T1 = 1. + VLIK(1)                                                  SRI 772
       DO 356 J = 1,NJ                                                    SRI 773
       T2 = T1 + VDPR(J) / FL                                             SRI 774
       FLKIJK = (T1 + T2 - T1/T2) / 3.                                    SRI 775
       FIJK = FLKIJK * DPR(J) * FL                                        SRI 776
       FKIJK = FL * T2 / (2. * FLKIJK)                                    SRI 777
       CSIJK = (CSZ(J) * FL * FLKIJK + (PJ(J) - XCI(J)))/XVDI(J)*VDPR(J) SRI 778
       BPIJK = (EXPF(DPR(J) * VDPR(J) / FIJK) -1.) / (DIPR * VDPR(J)) +1.SRI 779
       RPJK = FIJK * LOGF(FKIJK * (BPIJK ** QIPR - 1.) * (1. + CSIJK /    SRI 780
      1        (CHJJ(J) * FLKIJK * FLIK(1))) / STI)                       SRI 781
       RZJK = RPJK - (FMU + FMUK(1)) * FIJK                               SRI 782
```

```
      RZZJK = MINIF(RZJK, DPR(J) * TX)                                  SRI 783
      RJ1 = INTF(MAX1F(RZZJK, XJ1(J) + FK4 * DPR(J) *(STI+FLIK(1)))+.9) SRI 784
      RI = RI + SJ(J) * RJ1                                             SRI 785
      XRIJZ(J) = RZJK / DIJ(J)                                          SRI 786
C     ****** THE FOLLOWING STATEMENT IS FOR CHECKOUT ONLY *********     SRI 765
      PRINT 53,XXIJ(J), XXIJZ(J), XRIJZ(J), VDPR(J)
  356 CONTINUE
      RI = INTF(RI + .9)                                                SRI 787
      XI = RI - QI                                                      SRI 788
      SD   = SD   + DI                                                  SRI 789
      SDB  = SDB  + DBI                                                 SRI 790
      SGP  = SGP  + BG * DBI                                            SRI 791
      SGPP = SGPP + PG * DI                                             SRI 792
      SDLB = SDLB + DLBI                                                SRI 793
      SDL  = SDL  + DLI                                                 SRI 794
      SRP  = SRP  + RBI                                                 SRI 795
      SORD = SORD + 1. / STI                                            SRI 796
      SQ   = SQ   + DI * STI                                            SRI 797
      SQP  = SQP  + DBI * STI                                           SRI 798
      SR   = SR   + RI                                                  SRI 799
      RETURN                                                            SRI 800
C                                                                       SRI 801
C     PRINT SUMMARY                                                     SRI 802
C                                                                       SRI 803
    3 SG = SGP + SGPP                                                   SRI 804
      V2  = SRP / SDB                                                   SRI 805
      V3  = SR / SD                                                     SRI 806
      V4  = (SRP - SGP) / SDB                                           SRI 807
      V5  = (SR - SG) / SD                                              SRI 808
      W1  = SGP / SDB                                                   SRI 809
      W2  = SG / SD                                                     SRI 810
      V8  = SR - SDL                                                    SRI 811
      V9  = V8 - SQ / 2.                                                SRI 812
      V10 = SRP - SDLB                                                  SRI 813
      V11 = V10 - SQP / 2.                                              SRI 814
      PRINT 19,SD,V2,V3,V4,V5,W1,W2,SDLB,SDL,V8,V9,V10,V11,SORD         SRI 815
   19 FORMAT(1H0,14X,1HD,10X,5HRPR/D,12X,3HR/D,4X,11H(RPR-APR)/D,8X,7H(RSRI 816
     1-A)/D,10X,5HAPR/D,12X,3HA/D/7F15.3/1H0,10X,6H(DL)PR,11X,4H(DL),4X,SRI 817
     211H(ON-HND-MX),4X,11H(ON-HND-AV),4X,11H(ON-H-MX)PR,4X,11H(ON-H-AV)SRI 818
     3PR,12X,3HORD/7F15.3/1H0)                                          SRI 819
C     ****** THE FOLLOWING STATEMENT IS FOR CHECKOUT ONLY *********     SRI 820
   50 FORMAT (2H1A, 10F10.3)                                            SRI 821
   51 FORMAT (2H0B, 5F10.3)                                             SRI 822
   52 FORMAT (2H0C, 7F10.3)                                             SRI 823
   53 FORMAT (2H0D, 4F10.3)                                             SRI 824
   54 FORMAT (2H0E, 10F10.3 / 2H0 , 7F10.3)                             SRI 825
      END                                                               SRI 826
```

```
C
C
      SUBROUTINE PRO (ICON)                                             SRI 827
C                                                                       SRI 828
C     COMMON STORAGE                                                    SRI 829
C                                                                       SRI 830
      DIMENSION AO(26), BCCJK(20,25), BF(20,25), BPCJK(20,25),          SRI 831
     1          CFK(25), CJ(20), CPK(25), CPLK(25), CPPJ(20), DIJ(20),  SRI 832
     2          DIJK(20,25), DM(20,25,12), DWIJK(20,25), FLIK(25),      SRI 833
     3          FLSK(25), FMUK(25), FSK(25), FRK(25), OCJK(20,25),      SRI 834
     4          OHJK(20,25), OOJK(20,25), PJ(20), POAS(20),             SRI 835
     5          POAW(20), SIJK(20,25), SJ(20), UPCJ(20), WJ(20),        SRI 836
     6          XDIJ(20), XDIJK(20,25), XK2IK(20), XRIJK(20,25),        SRI 837
     7          XRIJZ(20), XXIJ(20), XXIJZ(20), XIJ(20)                 SRI 838
      COMMON    AO, A, APB, APP, AZR, BCCJK, BF, BG, BPCJK, CFK, CJ,    SRI 839
     1          CM2, CPK, CPLK, CPPJ, CR, DI, DIJ, DIJK, DM, DWIJK,     SRI 840
     2          FLIK, FLSK, FMU, FMUK, FLP, FLRI, FK1, FK2, FK3, FLC,   SRI
     3          FK4, FK5, FK6, FSK, FRK, J, K, NJ, NK, OCJK, ODD,       SRI 842
     4          OHJK, OOJK, ORD, PG, PJ, POBS, PODD, POAS, POAW, QI,    SRI 843
     5          SIJK, SJ, T, TX, UPCJ, WJ, XDIJ, XDIJK, XK2IK, XIJ,     SRI 844
     6          XRIJK, XRIJZ, XXIJ, XXIJZ                               SRI 845
C                                                                       SRI 846
      IF (ORD) 8,8,9                                                    SRI 847
    8 RETURN                                                            SRI 848
    9 IF (OHJK(J,1) - ORD) 8,1,1                                        SRI 849
    1 SIJK(J,K) = SIJK(J,K) + ORD                                       SRI 850
      OOJK(J,K) = OOJK(J,K) + ORD                                       SRI 851
      OHJK(J,1) = OHJK(J,1) - ORD                                       SRI 852
      SIJK(J,1) = SIJK(J,1) - ORD                                       SRI 853
      GO TO (6,7),ICON                                                  SRI 854
    6 OCJK(J,K) = OCJK(J,K) + 1.                                        SRI 855
    7 CFK(K) = CFK(K) + ORD * WJ(J) * FRK(K) * .01                      SRI 856
      CPK(K) = CPK(K) + ORD * WJ(J) * CR * .01                          SRI 857
      BPCJK(J,K) = BPCJK(J,K) + ORD - RNDLF(ORD, UPCJ(J) * CPPJ(J))     SRI 858A
      BCCJK(J,K) = BCCJK(J,K) + ORD - RNDLF(ORD, UPCJ(J))               SRI 858
      CPLK(K) = CPLK(K) + CPFP * RNDLF(ORD, UPCJ(J) * CPPJ(J))          SRI 859
      CALL QUEUEB(1)                                                    SRI 860
      END                                                               SRI 861
```

```
C
C
      SUBROUTINE PREDICT
C
C     COMMON STORAGE
C
      DIMENSION AO(26), BCCJK(20,25), BF(20,25), BPCJK(20,25),
     1          CFK(25), CJ(20), CPK(25), CPLK(25), CPPJ(20), DIJ(20),
     2          DIJK(20,25), DM(20,25,12), DWIJK(20,25), FLIK(25),
     3          FLSK(25), FMUK(25), FSK(25), FRK(25), OCJK(20,25),
     4          OHJK(20,25), OOJK(20,25), PJ(20), POAS(20),
     5          POAW(20), SIJK(20,25), SJ(20), UPCJ(20), WJ(20),
     6          XDIJ(20), XDIJK(20,25), XK2IK(20), XRIJK(20,25),
     7          XRIJZ(20), XXIJ(20), XXIJZ(20), XIJ(20)
      COMMON    AO, A, APB, APP, AZR, BCCJK, BF, BG, BPCJK, CFK, CJ,
     1          CM2, CPK, CPLK, CPPJ, CR, DI, DIJ, DIJK, DM, DWIJK,
     2          FLIK, FLSK, FMU, FMUK, FLP, FLRI, FK1, FK2, FK3, FLC,
     3          FK4, FK5, FK6, FSK, FRK, J, K, NJ, NK, OCJK, ODD,
     4          OHJK, OOJK, ORD, PG, PJ, POBS, PODD, POAS, POAW, QI,
     5          SIJK, SJ, T, TX, UPCJ, WJ, XDIJ, XDIJK, XK2IK, XIJ,
     6          XRIJK, XRIJZ, XXIJ, XXIJZ
C
      J1 = T + 44.
      J2 = T + 52.
      DI = C.
      DO 7 J = 1,NJ
      DIJ(J) = 0.
      DO 6 K = 1,NK
      IF (BF(J,K)) 6,6,1
    1 X = 0.
      DO 5 IT = J1,J2
      L = (IT - 1) * 12 / 52 - 11
      IF (L - 13) 3,3,2
    2 L = L - 12
      GO TO 5
    3 IF (L) 4,4,5
    4 L = L + 12
    5 X = X + DM(J,K,L) * 7. / 30.
      DIJK(J,K) = X / 9.
      DIJ(J) = DIJ(J) + X / 9.
    6 CONTINUE
    7 DI = DI + DIJ(J) * SJ(J)
      END
```

```
C
C
      SUBROUTINE RDTP                                                   SRI 903
C                                                                       SRI 009
C     COMMON STORAGE                                                    SRI 010
C                                                                       SRI 011
      DIMENSION AO(26), BCCJK(20,25), BF(20,25), BPCJK(20,25),         SRI 012
     1          CFK(25), CJ(20), CPK(25), CPLK(25), CPPJ(20), DIJ(20), SRI 013
     2          DIJK(20,25), DM(20,25,12), DWIJK(20,25), FLIK(25),      SRI 014
     3          FLSK(25), FMUK(25), FSK(25), FRK(25), OCJK(20,25),      SRI 015
     4          OHJK(20,25), OOJK(20,25), PJ(20), POAS(20),             SRI 016
     5          POAW(20), SIJK(20,25), SJ(20), UPCJ(20), WJ(20),        SRI 017
     6          XDIJ(20), XDIJK(20,25), XK2IK(20), XRIJK(20,25),        SRI 018
     7          XRIJZ(20), XXIJ(20), XXIJZ(20), XIJ(20)                 SRI 019
      COMMON    AO, A, APB, APP, AZR, BCCJK, BF, BG, BPCJK, CFK, CJ,    SRI 020
     1          CM2, CPK, CPLK, CPPJ, CR, DI, DIJ, DIJK, DM, DWIJK,     SRI 021
     2          FLIK, FLSK, FMU, FMUK, FLP, FLRI, FK1, FK2, FK3, FLC,   SRI
     3          FK4, FK5, FK6, FSK, FRK, J, K, NJ, NK, OCJK, ODD,       SRI 023
     4          OHJK, OOJK, ORD, PG, PJ, POBS, PODD, POAS, POAW, QI,    SRI 024
     5          SIJK, SJ, T, TX, UPCJ, WJ, XDIJ, XDIJK, XK2IK, XIJ,     SRI 025
     6          XRIJK, XRIJZ, XXIJ, XXIJZ                               SRI 026
C                                                                       SRI 027
C                                                                       SRI 906
      READ 3, (AO(K),K=1,4),(AO(K),K=13,25)                             SRI 907
      IF (AO(1)) 1,1,2                                                  SRI 908
    1 AO(1) = -1.                                                       SRI 909
    2 RETURN                                                            SRI 910
    3 FORMAT (F1.0,F4.0,F6.0,F3.0,F6.0,12F5.0)                          SRI 911
      END                                                               SRI 912
C
C
      SUBROUTINE GENDEM                                                 SRI 913
C                                                                       SRI 914
C     COMMON STORAGE                                                    SRI 915
C                                                                       SRI 916
C     (INSERT THE DIMENSION STATEMENT OF MAIN PROGRAM HERE)
C     (INSERT THE COMMON STATEMENT OF THE MAIN PROGRAM HERE)
C                                                                       SRI 932
      L = INTF((T - 1.) * 12. / 52. + 1.)                               SRI 933
      DO 2 J = 1,NJ                                                     SRI 934
      DO 2 K = 1,NK                                                     SRI 935
      IF (BF(J,K)) 2,2,1                                                SRI 936
    1 DWIJK(J,K) = INTF(DM(J,K,L) * 7. / 30. + .5)                      SRI 937
    2 CONTINUE                                                          SRI 938
      END                                                               SRI 939
```

```
C
      SUBROUTINE QUEUE3(N)                                              SRI 940
C                                                                       SRI 941
C     LOCAL STORAGE                                                     SRI 942
C                                                                       SRI 943
      DIMENSION QORD(150), QODD(150), QJ(150), QK(150)                  SRI 944
C                                                                       SRI 945
C     COMMON STORAGE                                                    SRI 946
C                                                                       SRI 947
      DIMENSION AO(26), BCCJK(20,25), BF(20,25), BPCJK(20,25),          SRI 948
     1          CFK(25), CJ(20), CPK(25), CPLK(25), CPPJ(20), DIJ(20),  SRI 949
     2          DIJK(20,25), DM(20,25,12), DWIJK(20,25), FLIK(25),      SRI 950
     3          FLSK(25), FMUK(25), FSK(25), FRK(25), OCJK(20,25),      SRI 951
     4          OHJK(20,25), OOJK(20,25), PJ(20), POAS(20),             SRI 952
     5          POAW(20), SIJK(20,25), SJ(20), UPCJ(20), WJ(20),        SRI 953
     6          XDIJ(20), XDIJK(20,25), XK2IK(20), XRIJK(20,25),        SRI 954
     7          XRIJZ(20), XXIJ(20), XXIJZ(20), XIJ(20)                 SRI 955
C     (INSERT THE COMMON STATEMENT OF THE MAIN PROGRAM HERE)
C                                                                       SRI 963
      GO TO (1,4,9),N                                                   SRI 964
    1 I = I + 1                                                         SRI 965
      IF (I - 150) 2,2,3                                                SRI 966
    2 QORD(I) = ORD                                                     SRI 967
      QODD(I) = ODD                                                     SRI 968
      QJ(I) = J                                                         SRI 969
      QK(I) = K                                                         SRI 970
      RETURN                                                            SRI 971
    3 PRINT 10                                                          SRI 972
      RETURN                                                            SRI 973
    4 IF (I) 7,7,5                                                      SRI 974
    5 DO 6 M = 1,I                                                      SRI 975
      IF (QODD(M) - T) 6,8,6                                            SRI 976
    6 CONTINUE                                                          SRI 977
    7 ORD = 0.                                                          SRI 978
      RETURN                                                            SRI 979
    8 ORD = QORD(M)                                                     SRI 980
      J = QJ(M)                                                         SRI 981
      K = QK(M)                                                         SRI 982
      QORD(M) = QORD(I)                                                 SRI 983
      QODD(M) = QODD(I)                                                 SRI 984
      QJ(M) = QJ(I)                                                     SRI 985
      QK(M) = CK(I)                                                     SRI 986
      QODD(I) = 0.                                                      SRI 987
      I = I - 1                                                         SRI 988
      RETURN                                                            SRI 989
    9 I = 0                                                             SRI 990
      RETURN                                                            SRI 991
   10 FORMAT (32H0    BRANCH ORDER QUEUE OVERFLOW)                      SRI 992
      END                                                               SRI 993
```

```
C
C
      SUBROUTINE QUEUE? (M)                                                SRI 994
C                                                                          SRI 995
C     LOCAL STORAGE                                                        SRI 996
C                                                                          SRI 997
      DIMENSION QPODD(10), QPOBS(10), QPOAW(10,20), QPOAS(10,20)           SRI 998
C                                                                          SRI 999
C     COMMON STORAGE                                                       SRI 000
C                                                                          SRI 001
      DIMENSION AO(26),  BCCJK(20,25), BF(20,25), BPCJK(20,25),            SRI 002
     1          CFK(25), CJ(20), CPK(25), CPLK(25), CPPJ(20), DIJ(20),     SRI 003
     2          DIJK(20,25), DM(20,25,12), DWIJK(20,25), FLIK(25),         SRI 004
     3          FLSK(25), FMUK(25), FSK(25), FRK(25), OCJK(20,25),         SRI 005
     4          OHJK(20,25), OOJK(20,25), PJ(20), POAS(20),                SRI 006
     5          POAW(20), SIJK(20,25), SJ(20), UPCJ(20), WJ(20),           SRI 007
     6          XDIJ(20), XDIJK(20,25), XK2IK(20), XRIJK(20,25),           SRI 008
     7          XRIJZ(20), XXIJ(20), XXIJZ(20), XIJ(20)                    SRI 009
C     (INSERT THE COMMON STATEMENT OF THE MAIN PROGRAM HERE)
C                                                                          SRI 017
      GO TO (1, 5, 9),M                                                    SRI 018
    1 DO 2 I = 1,10                                                        SRI 019
      IF (QPODD(I)) 2,3,2                                                  SRI 020
    2 CONTINUE                                                             SRI 021
      PRINT 11                                                             SRI 022
      RETURN                                                               SRI 023
    3 QPODD(I) = PODD                                                      SRI 024
      QPOBS(I) = POBS                                                      SRI 025
      DO 4 J = 1,NJ                                                        SRI 026
      QPOAS(I,J) = POAS(J)                                                 SRI 027
    4 QPOAW(I,J) = POAW(J)                                                 SRI 028
      RETURN                                                               SRI 029
    5 DO 6 I = 1,10                                                        SRI 030
      IF (QPODD(I) - T) 6,7,6                                              SRI 031
    6 CONTINUE                                                             SRI 032
      POBS = 0.                                                            SRI 033
      RETURN                                                               SRI 034
    7 POBS = QPOBS(I)                                                      SRI 035
      DO 8 J = 1,NJ                                                        SRI 036
      POAW(J) = QPOAW(I,J)                                                 SRI 037
    8 POAS(J) = QPOAS(I,J)                                                 SRI 038
      QPODD(I) = 0.                                                        SRI 039
      RETURN                                                               SRI 040
    9 DO 10 I = 1,10                                                       SRI 041
   10 QPODD(I) = 0.                                                        SRI 042
      RETURN                                                               SRI 043
   11 FORMAT (36H0    PRODUCTION ORDER QUEJE OVERFLOW)                     SRI 044
      END                                                                  SRI 045
```

```
C
C
      FUNCTION RNDNF (A,B)                                              SRI 046
      RNDNF = INTF (A / B + .50) * B                                    SRI 047
      END                                                               SRI 048
C
      FUNCTION RNDCF (A,B)                                              SRI 049
      RNDCF = INTF (A / B + .99) * B                                    SRI 050
      END                                                               SRI 051
C
      FUNCTION RNDLF (A,B)                                              SRI 052
      RNDLF = INTF(A / B) * B                                           SRI 053
      END                                                               SRI 054
      END
```

APPENDIX II

DICTIONARY OF VARIABLES

Variable Name		Definition	Unit
A		Production setup cost	Dollars/Setup
AO		Input data array	None
APB		Branch physical storage cost	Cents/Product unit/ Year
APP		Whs physical storage cost	Cents/Product unit/ Year
AZR		Opportunity cost	Cents/Dollar/Year
BASE		Product code number	None
BCCJK	J,K	Broken case count, item-location	Unit
BCK	K	Branch code	None
BCP		Broken ctn and pallets, system	Unit
BCPCJ	J	Broken ctn and pallet count, item	Unit
BCPCJK	J,K	Broken ctn and pallet count	Unit
BCPI		Broken ctn and pallets, product	Product Unit
BPCJK	J,K	Broken pallet count, item-location	Unit
BPIJK		Computational factor	None
BSI		Batch size increment	Product Unit
BSM		Batch size, minimum	Product Unit
BF	J,K	Branch usage flag	None
BG		Branch requisitioning goal	Days (of Supply)
BPCI		Broken pallet count, product	Unit
CBC		Cost of broken ctns, system	Dollars
CBCC		Cost of picking broken ctns	Dollars/Unit
CBCI		Cost of broken ctns, product	Dollars
CBCJ		Cost of broken ctns, item	Dollars
CBCJK		Cost of broken ctns, item-location	Dollars
CBP		Cost of broken pallets, system	Dollars
CBPC		Cost of picking broken pallets	Dollars/Unit
CBPI		Cost of broken pallets, product	Dollars
CBPJ		Cost of broken pallets, item	Dollars
CBPJK		Cost of broken pallets, item-location	Dollars
CBO		Cost of branch orders, system	Dollars
CBOK		Cost of branch orders, branch	Dollars
CF		Freight cost, system	Dollars
CFK	K	Freight cost, branch	Dollars/100 Pounds
CH		Holding costs, system	Dollars/Year
CHI		Holding costs, product	Dollars/Year
CHJ		Holding costs, item	Dollars/Year
CHJJ		Opportunity plus physical holding	Dollars/Unit/Week
CHJK		Holding costs, item-location	Dollars/Unit/Week
CHK	K	Holding cost rate, branch	Dollars/Unit/Week
CIJ1		Whse unit cost	Dollars/Unit
CIJ	K	Branch unit cost	Dollars/Unit
CJ	J	Production unit cost	Dollars/Unit
CL		Cost of losses, system	Dollars
CLI		Cost of losses, product	Dollars

CLJ		Cost of losses, item	Dollars
CLJK		Cost of losses, item-location	Dollars
CLK	K	Cost of losses, branch total	Dollars
CKZJK		Variance factor	None
CM2		Order rate multiplier	None
CP		Cost of setups, system	Dollars
CPD		Cost of loss per demand, system	Dollars/Unit
CPDK		Cost of loss per demand, branch	Dollars/Unit
CPP		Cost of order preparation and process, system	Dollars
CPPK		Cost of order preparation and process, branch	Dollars
CPK	K	Processing cost, branch	Dollars
CPFP		Cost of picking full pallet, rate	Dollars/Pallet
CPLK	K	Cost of picking and loading, branch	Dollars
CPPJ		Cartons per pallet	Ctns/Pallet
CPPO		Order preparation and processing cost, rate	Dollars/Order
CPIJK	J,K	Processing cost, item-location	Dollars
CR		Cost of receiving (processing) per unit	Dollars/100 Pounds
CSIJK		Profit (selling-branch cost) also cost of shortages for both direct shortages at branches and shortages resulting from stockout at the central warehouse	Dollars/Unit
CTOT		Sum of CPL, CP, CL, SH, SYS-TOT	Dollars
CSZ	J	Cost of losses, item	Dollars
CPL		Cost of picking, loading, system	Dollars
DBI		Demand rate, branches less warehouse	Product unit/Week
DBIJJ		Demand rate, branches less warehouse	Unit/Week
DI		Predicted demand, product	Product unit/Week
DIJ	J	Predicted demand, item	Unit/Week
DIJK	J,K	Predicted demand, item-location	Unit/Week
DICHI		Mean holding cost/unit of product times system demands(DI)	Dollars/Product unit
DIPR		Mean demand between orders, product	Unit
DIPRPR		Mean demand between orders, item	Unit
DLBI		Leadtime demands, branches less whse, product	Product unit
DLBJ		Leadtime demands, branches less whse, item	Unit
DLI		Leadtime demands, whse, product	Product unit
DLIJ		Leadtime demands, whse, item	Unit
DM	J,K,L	Monthly demands	Unit
DPR	J	Probable demand in half of leadtime	Unit
DSCAJ		Discrete adjustment interval	Week
DSIJ		Satisfied demand, item	Unit
DSIJK	J,K	Satisfied demand, item-location	Unit
DWIJK	J,K	Weekly demands, item-location	Unit
DZ		Sum of DZIJK over all products	Unit
DZIJK		Mean of distribution for number of demands in prodcn cycle plus leadtime	None

F		Correction factor to Wilson's EOQ; Sum of all FIJK	Product unit
F2		Square of F	Product unit squared
FITEM	J	Item number, floating point mode	None
FIJK		The demands in a production leadtime adjusted for the variances of the demand and leadtime distributions	Unit
FK1		Multiplier, branch requisition objective floor	None
FK2		Multiplier, branch reorder level floor	None
FK3		Multiplier, warehouse requisition objective floor	None
FK4		Multiplier, warehouse reorder level floor	None
FK5		Multiplier, item production trigger level	None
FK6		Multiplier, production trigger level, warehouse	None
FKIJK		Variance factor for demands and production reorder leadtime; used to compute RIJK(J,K) for branches other than warehouse	None
FKZJK		Same as FKIJK except applies to branch interim reorder leadtime; used to compute RIJK(J,1)	None
FL		Warehouse production leadtime; for central whse LC + LP + LRI/2	Week
FLC		Communication leadtime	Week
FLIK	K	Branch production leadtime; Sum of LRI/2,LC,LP,LSK,LW/2	Week
FLDI		Maximum EOQ due to shelf life limitations	Product unit
FLP		Production-run leadtime	Week
FLPDTM		Time of placing most recently delivered production allocation	A particular week
FLQI		Optimal ratio of EOQ to square root of product demands	Unit $^{\frac{1}{2}}$
FLR		Reaction leadtime	Week
FLRI		Review interval	Week
FLKIJK		Variance correction factor for demands and production leadtime to be applied to production leadtime demands	None
FLZK	K	Branch interim reorder leadtime; LSK + LRI/2 + LW/2	Week
FLS		Leadtime, shipping, mean	Week
FLSK	K	In-transit leadtime, branch	Days
FMU		System capacity constraint	None
FMUK	K	Branch capacity constraint	None
FMDF		Demand forecast multiplier	None
FMBG		Branch goals, multiplier	None
FMPG		Plant goals, multiplier	None

FMCJ		Standard cost multiplier	None
FMLP		Leadtime multiplier, plant to warehouse	None
FMLWB		Leadtime multiplier, warehouse to branch	None
FNOPR		Number of production runs, floating point	None
FRK	K	Freight rate	Cents/Cwt
FZJK		Demands in branch interim order leadtime	Unit
GENDEM		Subroutine to generate demands	None
HC		Holding cost due to rounding to pallets	Dollars
ICON		Parameter for RXCOMP subroutine	None
ICON1		Rule selector; 1-SRI, 2-other	None
ICON2		Not used	None
ICON3		Stockout policy selector, 1- issue to zero balance 2- do not make partial issues	None
ICON4		Round to cartons selector, 1- do not round to full case lot 2- round to next higher case lot	None
ICON5		Round to pallets selector, 1- do not 2- round if holding cost is less than handling costs	None
I		Index	None
IEOF		End of file, fixed point mode	None
IT		Index	None
J		Index for item	None
J1		Index used in subroutine predict	None
J2		Index used in subroutine predict	None
K		Index for branch	None
L		Index for month and production order	None
M		Index and parameter for subroutine QUEUEP	None
N		Index parameter for subroutine QUEUEB	None
NJ		Number of items	None
NK		Number of branches	None
OC		Order count, system	Orders
OCI		Order count, product	Orders
OCJ		Order count, item	Orders
OCJK	J,K	Order count, item-location	Orders
ODD		Order delivery date	Week
OHJK	J,K	On-hand, item-location	Unit
FSK	K	Shipping frequency	Week
OOJK	J,K	On order, item-location, cumulative	Unit
ORD		Order quantity	Unit
OCK		Order count, branch, mean	Orders
PAGE		Page number	None
PBO		Subroutine to place branch orders	None
PG		Plant goal	Days (of Supply)
PJ	J	Unit selling price, branch	Dollars/Unit
POAS	J	Production order allocation to item	Unit

POAW	J	Production order allocation to warehouse	Unit
POBS	J	Production batch size	Product unit
PODD		Production order delivery date	Week
PC		Picking cost due to roundg to cartons	Dollars
PRIJK		Probability of order between production allocations	None
PREDICT		Subroutine to predict demands	None
QI		Economic prodcn order qty	Product unit
QIZ		Computational factor for QI	Product unit
QIZZ		Minimum of alternate values of EOQ	Product unit
QIPR		Computational factor for RIJK, XIJK	Product unit
QIPR		The number of orders corresponding to the adjusted EOQ quantity	None
QJ	I	Order item index in subroutine queue	None
QK	I	Order branch index in subroutine queue	None
QORD	I	Order quantity in subroutine queue	Unit
QODD	I	Order delivery date in subroutine queue	A specific week
QUEUEB	N	Subroutine for queuing branch orders	None
QPOAS	J,I	Production item allocatn in queue	Unit
QPOAW	J,I	Production whse alloctn in queue	Unit
QPOBS	I	Production batch size in queue	Product unit
QPODD	I	Production order date in queue	A specific week
QUEUEP	M	Subroutine for queuing production order	None
QRIJK		Modified EOQ value	Product unit
R		Scaling factor; POAS/ORD	None
RBI		Requisition objective level, branch only	Product unit
RGX		Ratio for alternate trigger level	None
RI		Requisition objective level (goal), product	Product unit
RIJ		Requisition objective level (goal), item	Unit
RIJJ		Requisition objective level (goal), item	Unit
RIJK	J,K	Requisition objective level (goal), item-location	Unit
RJK		Maximum of RZZJK and XIJK plus lead-time demands	Unit
RJ1		Requisition objective level, warehouse only, item	Unit
RLT		Ratio of losses to total demands, system	None
RLTI		Ratio of losses to total demands, product	None
RLTJ		Ratio of losses to total demands, item	None
RLTJK		Ratio of losses to total demands, item-location	None

RPJK		Computational factor for requisition objective	Unit
RST		Stockturn ratio, system	None
RSTI		Stockturn ratio, product	None
RSTJ		Stockturn ratio, item	None
RSTK		Stockturn ratio, branch	None
RSTJK		Stockturn ratio, item-location	None
RUNNO		Run number	None
RXCOMP		Subroutine to compute SRI rules	None
RZJK		Alternate value of requisition objective	Unit
RZZJK		Minimum of alternate values of RIJK	Unit
RDTP		Subroutine to read input data	None
SC	J	Size (item) code	None
SD		System demands	Product unit
SDB		System demds less warehouse demds	Product unit
SDL		System demds in leadtime	Product unit
SDLB		Sys demds in leadtime, branches only	Product unit
SG		System goal all products, and locations	Product unit
SGP		System goal, less warehouse	Product unit
SGPP		System goal, warehouse	Product unit
SI		Inventory position, product	Product unit
SIJ	J	Inventory position, item	Unit
SIJK	J,K	Inventory position, item-location	Unit
SJ	J	Product unit conversion factor	Product unit/Unit
SOC		Stockout count, system	None
SOCI		Stockout count, product	None
SOCJ		Stockout count, item	None
SOCJK	J,K	Stockout count, item-location	None
SORD		Cumulative ord qty	Product unit
SOW		Stockout weeks, system	Week
SOWI		Stockout weeks, product	Week
SOWJ		Stockout weeks, item	Week
SOWJK	J,K	Stockout weeks, item-location	Week
SQ		Economic order quantity, system	Product unit
SQI		Economic order quantity, product	Product unit
SQP		Economic order quantity, branches only	Product unit
SQWI2		Square of Wilson's EOQ, product	Product unit
SR		System requisition objective, all prods + locs	Product unit
SRP		System requisition objective, branches only	Product unit
STI		Coverage factor, product	Week
STIK		Coverage factor, location	Week
SY		Average inventory position, system	Product unit
SYI		Average inventory position, product	Product unit
SYJ		Average inventory position, item	Unit
SYJK	J,K	Average inventory position, item-location	Unit
SOWK	K	Stockout weeks, branch	Week
T		Time index	None

TI		Coverage provided by QI (BAR) plus one half of reaction time	Week
TP		Temporary storage location for OHJK	None
TP2		Temporary storage location for DSIJK	None
TX		Shelf life	Month
T12		Interval between time of placing most recently delivered production allocation and present time	Week
UL		Units loss, system	Product unit
ULI		Units loss, product	Product unit
ULJ		Units loss, item	Unit
ULJK		Units loss, item-location	Units
UPCJ	J	Units per carton	Units/Carton
VARDF		Function to compute variance	None
VDIJ1		Rel-variance of demands, whs	None
VDPR	J	Probability of interim order	None
VDIJK		Rel-variance of demand, item-location	None
VDS		Value of demand satisfied, system	Dollars
VDSI		Value of demand satisfied, product	Dollars
VDSJ		Value of demand satisfied, item	Dollars
VDSK	K	Value of demand satisfied, branch	Dollars
VDSJK		Value of demand satisfied, item-location	Dollars
VLP		Rel-variance of prodcn leadtime	None
VLZK		Rel-variance of branch leadtime	None
VSIT		Value of inventory in-transit	Dollars
VSYI		Value of ave inventory, product	Dollars
VSYJ		Value of ave inventory, item	Dollars
VSYK	K	Value of ave inventory, branch	Dollars
VSYJK		Value of ave inventory, item-location	Dollars
VSY		Value of ave inventory, system	Dollars
VYSIT		Value of ave inventory in-transit	Dollars
VZD		Value of total demands, system	Dollars
VZDI		Value of total demands, product	Dollars
VZDJ		Value of total demands, item	Dollars
VZDK	K	Value of total demands, branch	Dollars
VLIK	K	Rel-Variance of branch leadtime	None
VDZJK		Rel-variance of demands, item-location	None
V2		Control knob - system requisition level in weeks of demand except for central whse under SRI rules	None
V3		Control knob - system requisition level in weeks of demand including central whse under SRI rules	None
V4		Control knob - difference between V2 and W1	None
V5		Control knob - difference between V3 and W2	None
V8		Control knob - expected maximum on hand level, all branches	None
V9		Control knob - expected average on hand level, all branches	None

V10		Control knob - same as V8 except warehouse is excluded	None
V11		Control knob - same as V9 except warehouse is excluded	None
W		Temp storage location	None
WJ	J	Unit weight	Pounds/Unit
WPSO		Weeks per stockout, system	Week
WPSOI		Weeks per stockout, product	Week
WPSOJ		Weeks per stockout, item	Week
WPSOJK		Weeks per stockout, item-location	Week
W1		System requisition level in weeks of demand except for central whse under alternate rules	Week
W2		System requisition level in weeks of demand including central whse under alternate rules	Week
X		Temporary storage location	None
XCI	J	Dealer const (unit plus shipping plus receiving)	Dollars/Unit
XDIJ	J	Demands over shelf life, item	Unit
XDIJK	J,K	Demands over shelf life, item-location	Unit
XI		Reorder trigger level, product	Product unit
XIJ	J	Reorder trigger level, item	Unit
XIJK	J,K	Reorder trigger level, item-location	Unit
XJ1	J	Reorder trigger level, warehouse	Unit
XJK		Reorder trigger level, item-location other than warehouse	Unit
XK2IK	J	Interim order leadtime times reorder floor multiplier (FK2)	None
XRIJK	J,K	Requisition objective level at a branch in weeks of item location demands	Week
XRIJZ	J	Requisition objective level at warehouse in weeks of total item demand	Week
XXIJ	J	Reorder level for item in weeks of total item demand	Week
XXIJZ	J	Reorder level for warehouse in weeks of total item demand	Week
Y		Temporary storage location	None
ZDIJ		Total demands item	Unit
ZDIJK	J,K	Total demands item-location	Unit
ZVC		Total of CH, OC, CL; system	Dollars
ZVCK		Total of CHK, CBOK, CLK; branch	Dollars
Z		Obsolescence cost (not used in present form of simulator)	Cents/Unit

"Proprietary status inquiry ([illegible]' 9/22/60

[redacted]